

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**

THIS PAGE BLANK (USPTO)

(19)



Europäisches Patentamt
European Patent Office
Office européen des brevets



(11) Publication number:

0 543 599 A2

(12)

EUROPEAN PATENT APPLICATION

(21) Application number: 92310438.4

(51) Int. Cl.⁵: G06K 9/20

(22) Date of filing: 16.11.92

(30) Priority: 19.11.91 US 794275

(43) Date of publication of application:
26.05.93 Bulletin 93/21(64) Designated Contracting States:
DE FR GB(71) Applicant: XEROX CORPORATION
Xerox Square
Rochester New York 14644(US)(72) Inventor: Bloomberg, Dan S.
1013 Paradise Lane
Palo Alto, California 94306(US)
Inventor: Lakshmi, Dasari
3375 Alma Street No. 383
Palo Alto, California 94304(US)(74) Representative: Johnson, Reginald George et al
Rank Xerox Patent Department, Albion
House, 55-59 New Oxford Street
London WC1A 1BS (GB)

(54) Method and apparatus for image hand markup detection.

(57) An image markup detection device and method identifies and extracts markup lines and regions marked automatically or interactively by a user with an ordinary pen or pencil. Only morphological image processing operations on a scanned source image are used, resulting in the extrapolation of markup lines and marked regions. The markup lines are either extracted from the image, or the background information of the image (e.g., text) is removed, leaving only the markup lines. The marked region can then be printed, transferred or otherwise processed.

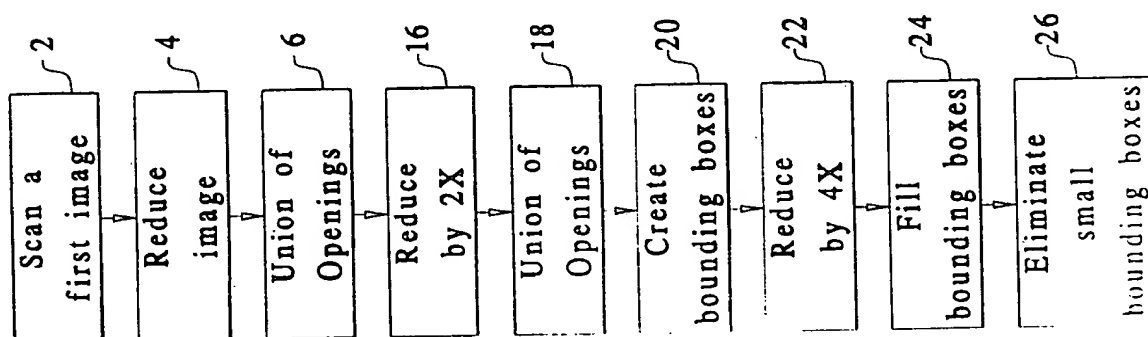


Fig. 2

EP 0 543 599 A2

1992

5

10

identifying parts of an image that are to be stored as a bitmap image; and

Identification of certain portions of a document for image processing has been accomplished in the prior

15

25

35

40

45

image to which the hand marks refer; and reproducing the identified regions of the scanned first image to which the hand marks refer without interference from the hand marks.

In one embodiment step (a) further comprises steps of reducing the first image by two times, using a threshold of 1; and taking a UNION of the first set of structuring elements with the second image, forming a third image.

In another embodiment some of the bounding boxes having less than a predetermined size are small bounding boxes, wherein the small bounding boxes are eliminated.

In one embodiment there is provided a method for processing a scanned first image in a digital computer to identify a location of non-transparent hand marks and marked regions, characterised by

- (a) CLOSING the first image with a first horizontal structuring element, forming a second image;
- (b) XORing the first image with the second image, forming a third image;
- (c) DILATING the third image with a solid square structural element, ANDing the DILATION of the third image with the first image, forming a fourth image;
- (d) XORing the fourth image with the first image, forming a fifth image;
- (e) taking a union of multiple OPENINGS of a first set of structuring elements and the fifth image, forming a sixth image;
- (f) OPENING the sixth image with a second horizontal structuring element, forming a seventh image;
- (g) reducing the seventh image by a first predetermined factor, forming an eighth image;
- (h) CLOSING the eighth image with the first set of structuring elements, forming a ninth image;
- (i) taking a union of OPENINGS of the ninth image with the first set of structuring elements, forming a tenth image;
- (j) reducing the tenth image by a second predetermined factor, forming an eleventh image;
- (k) filling the bounding boxes of the eleventh image, forming a twelfth image;
- (l) expanding the twelfth image to a full scale; and
- (m) ANDing the twelfth image with the first image, extrapolating the hand marked region.

In another embodiment there is provided a topological method for processing an image in a digital computer for extraction of regions of a document image encircled by non-transparent hand marks, characterised by

- (a) flood filling the document image from edges;
- (b) bitwise inverting the document image;
- (c) flood filling the document image from edges; and
- (d) bitwise inverting the document image.

Another aspect of the invention is the provision of a semitopological method for processing an image in a digital computer for extracting regions of a document image encircled by non-transparent hand marks, characterised by

- (a) flood filling the document image from edges;
- (b) bitwise inverting the document image;
- (c) OPENING the document image using a solid structuring element of a first predetermined size;
- (d) CLOSING the document image using a solid structuring element of a second predetermined size; and
- (e) ANDing the CLOSED document image with the original document image.

The first predetermined size may be a size less than a size of a character or at least as large as a size of a character.

In one aspect of the invention there is provided a method for processing a document image in a digital computer for identification in the document image of hand drawn lines comprising the steps of:

- (a) OPENING the document image using at least one structuring element;
- (b) finding bounding boxes around image units in the document image; and
- (c) testing the document image to identify horizontal hand drawn lines.

In one embodiment step (a) is preceded by a step of reducing the document image and includes using at least one horizontal or vertical structuring element so that horizontal or vertical lines are identified.

In another aspect of the invention there is provided a method for processing a document image in a digital computer for identifying hand drawn encircled regions of the document image, the method comprising the steps of:

- (a) OPENING the document image;
- (b) finding bounding boxes of image units in the document image;
- (c) testing the document image to identify hand drawn lines; and
- (d) filling into the original document image as a result of OPENING the document image, and filling into the original image.

In one embodiment step (a) is preceded by a step of reducing the document image and/or includes OPENING the image using horizontal or vertical structuring elements.

In one embodiment the method includes finding bounding boxes of the encircled regions; and extracting the encircled regions from the original document image so as to identify contents of the encircled regions.

5 The method may also include XORing a result of step (d) with a result of step (f) so as to extract the contents of the circled regions.

In yet another aspect of the invention there is provided an apparatus for processing a scanned first image in a digital computer for differentiating machine marks from hand marks and identifying a location of non-transparent hand marks and hand marked regions, characterised by

10 image scanning means for scanning a first image;

processor means for identifying and differentiating markings on the scanned first image as hand and machine marks using characteristics of the markings, said characteristics including horizontal, vertical, oblique, curved and irregular shapes;

identification means for identifying regions of the scanned first image to which the hand marks refer;

15 reducing means for reproducing the identified regions of the scanned first image to which the hand marks refer without interference from the hand marks; and

an output means for outputting the reproduction of the identified regions.

In one embodiment of the invention there is provided a method and apparatus for image markup detection of hand marks which work on binary scanned images and which utilize binary morphological image processing techniques to expedite the detection process.

20 In another embodiment of the invention there is provided a method and apparatus for image markup detection of hand marks which utilize detection without the use of either a highlighter pen or a gray-scale scanner.

One of the advantages provided by embodiments of the invention is the provision of methods and apparatus for image markup detection of hand marks which do not require extraneous detection circuitry to operate properly.

The present invention will be described further, by way of examples, with reference to the accompanying drawings in which like reference numerals refer to like elements. The drawings are not drawn to scale, rather, they illustrate the sequential image processing of a scanned first image according to various 30 embodiments of the invention.

Figure 1 is a flowchart of a first preferred embodiment according to the invention showing the steps in a direct approach to a method of identification and extraction of hand markup lines in an optically scanned first image;

35 Figure 2 is a flowchart of a second preferred embodiment according to the invention showing the steps of another direct approach to a method of identification and extraction of hand markup lines in an optically scanned first image;

Figure 3 is a flowchart utilizing binary logic symbols for a third preferred embodiment according to the invention showing the steps of an indirect approach to a method of identification and extraction of hand markup lines in an optically scanned first image;

40 Figure 4 is a block diagram of a preferred embodiment of an apparatus which identifies and extracts hand markup lines in an optically scanned first image according to the invention;

Figure 5 is an example of a scanned first image of a page of a document with markup lines hand drawn with an ordinary pen;

45 Figure 6 is an example of a second image formed by a first preferred embodiment of the indirect markup detection method of Fig. 3, the second image resulting from a morphological CLOSING operation of the first image of Figure 5;

Figure 7 is an example of a third image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the fourth image resulting from XORing the first image of Figure 5 with the second image of Figure 6;

50 Figure 8 is an example of a fourth image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the fourth image resulting from logically ANDing the DILATION of the third image of Figure 7 with the first image of Figure 5;

Figure 9 is an example of a fifth image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the fifth image resulting from XORing the fourth image of Figure 8 with the first image of Figure 5;

Figure 10 is an example of a sixth image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the sixth image resulting from an UNION of OPENINGS of the fifth image of Figure 9;

Figure 11 is an example of a seventh image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the seventh image resulting from morphological OPENINGS of the sixth image of Figure 10;

5 Figure 12 is an example of an eighth image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the eighth image resulting from a reduction of the seventh image of Figure 11 by a factor of four;

Figure 13 is an example of a ninth image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the ninth image resulting from the CLOSING of the eighth image of Figure 12;

10 Figure 14 is an example of a tenth image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the tenth image resulting from an UNION of OPENINGS on the ninth image of Figure 13;

Figure 15 is an example of an eleventh image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the eleventh image resulting from a reduction of the tenth image of Figure 14 by a factor of two;

15 Figure 16 is an example of a twelfth image (of the first preferred embodiment) formed by the indirect hand markup detection method of Fig. 3, the twelfth image resulting from filling the bounding boxes of the eleventh image of Figure 15;

Figure 17 is an example of a second image formed by the direct hand markup detection method of Fig. 1, the second image resulting from an UNION of OPENINGS of the first image of Figure 5;

20 Figure 18 is an example of a third image formed by the direct hand markup detection method of Fig. 1, the third image resulting from a reduction and UNION of OPENINGS of the second image of Figure 17;

Figure 19 is an example of a fourth image formed by the direct hand markup detection method of Fig. 2, the fourth image resulting from a reduction of the third image of Figure 18;

25 Figure 20 is an example of a fifth image (mask) formed by the direct hand markup detection method of Fig. 2, the fifth image resulting from a bounding box fill of the fourth image of Figure 19;

Figure 21 exemplifies a horizontal structuring element of length 8;

Figure 22 exemplifies a horizontal structuring element of length 2;

Figure 23 exemplifies a horizontal structuring element of length 5;

30 Figure 24 exemplifies a 5 x 5 structuring element with ON pixels running diagonally from the lower left corner to the upper right corner;

Figure 25 exemplifies a 5 x 5 structuring element with ON pixels running diagonally from the upper left corner to the lower right corner;

Figure 26 exemplifies a vertical structuring element of length 5;

35 Figures 27A - H exemplify a set of eight structuring elements of various configurations, each of length 9;

Figure 28 is an example of a first image of the second preferred embodiment of a direct approach for hand markup detection;

Figure 29 is an example of a second image of the second preferred embodiment of a direct approach for hand markup detection;

40 Figure 30 is an example of a third image of the second preferred embodiment of a direct approach for hand markup detection;

Figure 31 is an example of a fourth image of the second preferred embodiment of a direct approach for hand markup detection;

45 Figure 32 is an example of a fifth image of the second preferred embodiment of a direct approach for hand markup detection;

Figure 33 is an example of a sixth image of the second preferred embodiment of a direct approach for hand markup detection;

Figure 34 is an example of a seventh image of the second preferred embodiment of a direct approach for hand markup detection;

50 Figure 35 is an example of a first image of the third preferred embodiment of an indirect approach for hand markup detection;

Figure 36 is an example of a second image of the third preferred embodiment of an indirect approach for hand markup detection;

55 Figure 37 is an example of a third image of the third preferred embodiment of an indirect approach for hand markup detection;

Figure 38 is an example of a fourth image of the third preferred embodiment of an indirect approach for hand markup detection;

Figure 39 is an example of a fifth image of the third preferred embodiment of an indirect approach for hand markup detection;

Figure 40 is an example of a sixth image of the third preferred embodiment of an indirect approach for hand markup detection;

5 Figure 41 is an example of a seventh image of the third preferred embodiment of an indirect approach for hand markup detection;

Figure 42 is an example of a eighth image of the third preferred embodiment of an indirect approach for hand markup detection;

10 Figure 43 is an example of a ninth image of the third preferred embodiment of an indirect approach for hand markup detection;

Figure 44 is a flowchart of a topological method for extraction of regions of a document image encircled by non-transparent hand marks

Figure 45 is a flowchart of a semi-topological method for extracting regions of a document image encircled by non-transparent hand marks; and

15 Figure 46 is a flowchart of a method for extracting regions of a document image encircled by non-transparent hand marks.

I. Definitions and Terminology

20 The present discussion deals with binary images. In this context, the term "image" refers to a representation of a two-dimensional data structure composed of pixels. A binary image is an image where a given pixel is either "ON" or "OFF". Binary images are manipulated according to a number of operations wherein one or more source images are mapped onto a destination image. The results of such operations are generally referred to as images. The image that is the starting point of processing will sometimes be referred to as the original image or source image.

25 A "morphological operation" refers to an operation on a pixelmap image (a "source" image), that uses a local rule at each pixel to create another pixelmap image, the "destination" image. This rule depends both on the type of the desired operation to be performed as well as on the chosen "structuring element".

30 Pixels are defined to be ON if they are black and OFF if they are white. It should be noted that the designation of black as ON and white as OFF reflects the fact that most documents of interest have a black foreground and a white background. The techniques of the present invention could be applied to negative images as well. The discussion will be in terms of black on white, but the references to ON or OFF apply equally well to images which have been inverted and, therefore, the roles of these two states are reversed. In some cases the discussion makes reference to a "don't care" pixel which may be either an ON or an

35 OFF pixel.

A "structuring element" (SE) refers to an image object of typically (but not necessarily) small size and simple shape that probes the source image and extracts various types of information from it via the chosen morphological operation. In the attached figures that show Ses, a solid circle is a "hit", and an open circle is a "miss". The center position is denoted by a cross. Squares that have neither solid nor open circles are "don't cares"; their value in the image (ON or OFF) is not probed. A binary SE is used to probe binary images in a binary morphological operation that operates on binary input images and creates an output binary image. The SE is defined by a center location and a number of pixel locations, each normally having a defined value (ON or OFF). The pixels defining the SE do not have to be adjacent each other. The center location need not be at the geometrical center of the pattern; indeed it need not even be inside the pattern.

40 A "solid" SE refers to an SE having a periphery within which all pixels are ON. For example, a solid 2x2 SE is a 2x2 square of ON pixels. A solid SE need not be rectangular. A horizontal SE is generally one row of ON pixels and a vertical SE is generally one column of ON pixels of selected size. A "hit-miss" SE refers to an SE that specifies at least one ON pixel and at least one OFF pixel.

AND, OR and XOR are logical operations carried out between two images on a pixel-by-pixel basis.

50 NOT is a logical operation carried out on a single image on a pixel-by-pixel basis.

"EXPANSION" is a scale operation characterized by a scale factor N, wherein each pixel in a source image becomes an N x N square of pixels, all having the same value as the original pixel.

60 "REDUCTION" is a scale operation characterized by a scale factor N in a threshold level M. Reduction with scale = N entails dividing the source image into N x N squares of pixels, mapping each such square in the source image to a single pixel on the destination image. The value for the pixel in the destination image is determined by the threshold level M, which is a number between 1 and N². If the number of ON pixels in the pixel square is greater or equal to M, the destination pixel is ON, otherwise it is OFF.

"EROSION" is a morphological operation wherein a given pixel in the destination image is turned ON if and only if the result of superimposing the SE center on the corresponding pixel location in the source image results in a match between all ON and OFF pixels in the SE and the underlying pixels in the source image. An EROSION will give one pixel in the destination image for every match. That is, at each pixel, it
 5 outputs 1 if the SE (shifted and centered at that pixel) is totally contained inside the original image foreground, and outputs 0 otherwise. Note that EROSION usually refers to operations using a structuring element(s) with only hits, and more generally, matching operations with both hits and misses (often called a "hit-miss transform". The term EROSION is used herein to include matching operations with both hits and misses, thus the hit-miss transform is the particular type of EROSION used herein.

10 "DILATION" is a morphological operation wherein a given pixel in the source image being ON causes the SE to be written into the destination image with the SE center at the corresponding location in the destination image. The SEs used for DILATION typically have no OFF pixels. The DILATION draws the SE as a set of pixels in the destination image for each pixel in the source image. Thus, the output image is the union of all shifted versions of the SE translated at all 1-pixels of the original image.

15 A "seed fill" is an operation taking as input two images, and generating a third image as the result. One of the input images is the "seed", which may be composed of a single ON pixel or of many ON pixels. The other input image is the "mask", which is typically composed of more than one image component. The two images are aligned. The result of the seed fill is to produce an image that has only those image components in which at least one seed pixel was present in the seed image. The result image is formed by
 20 starting with the seed pixels and growing each image regions until it has filled the corresponding image component in the mask. This can be done morphologically (the "fillclip" operation, where the result image is formed by starting with the seed and alternatively dilating it and ADDING it with the "mask", until it stops changing) or by seed fill or "flood fill" techniques (where those image components containing a seed are erased -- by converting ON pixels to OFF pixels -- and then reconstructed using XOR with the original
 25 image).

"FillClip" is a morphological operation where one image is used as a seed and is grown morphologically, clipping it at each growth step to the second image. For example, a fillClip could include a DILATION followed by logically ANDing the DILATION result with another image.

30 "OPENING" is a morphological operation that uses an image and a structuring element and consists of an EROSION followed by a DILATION. The result is to replicate the structuring element in the destination image for each match in the source image.

"CLOSING" is a morphological operation using an image and a structuring element. It includes a DILATION followed by an EROSION of the image by a structuring element. A CLOSE of an image is equivalent to the bit inverse of an OPEN on the (bit inverse) background.

35 An UNION is a bitwise OR between two images. An "intersection" is a bitwise AND between two images.

"BLURRING" is a DILATION using a structuring element(s) composed of two or more hits.

40 A "mask" refers to an image, normally derived from an original or source image, that contains substantially solid regions of ON pixel corresponding to regions of interest in the original image. The mask may also contain regions of ON pixels that do not correspond to regions of interest.

"Text" refers to portions of a document or image which comprises letters, numbers, or other language symbols including non-alphabetic linguistic characters such as ideograms and syllabry in the oriental languages.

45 The various operations defined above are sometimes referred to in noun, adjective, and verb forms. For example, references to DILATION (noun form) may be in terms of DILATING the image or the image being DILATED (verb forms) or the image being subjected to a DILATION operation (adjective form). No difference in meaning is intended.

Morphological operations have several specific properties that simplify their use in the design of appropriate algorithms. First, they are translationally invariant. A sideways shift of the image before
 50 transforming does not change the result, except to shift the result as well. Operations that are translationally invariant can be implemented with a high degree of parallelism, in that each point in the image is treated using the same rule. In addition, morphological operations satisfy two properties that make it easy to visualize their geometrical behavior. First, EROSION, DILATION, OPEN and CLOSE are "increasing", which means that if image 1 is contained in image 2, then many of these morphological operations on image 1 will
 55 also be contained in the morphological operation on image 2. Second, a CLOSE is extensive and OPEN is antiextensive. This means that the original image is contained in the image transformed by CLOSE and the image transformed by OPEN is contained in the original image. The DILATION and EROSION operations are also extensive and antiextensive, respectively, if the center of the structuring element is contained within

the original image.

The OPEN and CLOSE operations also satisfy two more morphological properties:

- (1) The result of the operation is independent of the position of the center of the structuring element.
- (2) The operation is idempotent, which means that reapplying the OPEN or CLOSE to the resulting image will not change it.

An "image unit" means an identifiable segment of an image such as a word, number, character, glyph or other unit that can be extracted reliably and have an underlying linguistic structure.

II. Overview of the Method

One problem addressed by the invention is identifying regions (i.e., image segments) on a page that have been marked by hand with an ordinary pen (or pencil). The markings can consist of horizontal or vertical lines, or of "circular" marks (either open lines, segments of curved lines, or combinations of the two). Since all markings are by hand, the straight lines will not be the same straightness or smoothness of machine - printed rules, or hand markings using a straight - edge.

The image interpretation problem can be broken into several sub - problems;

- (1) identifying the markings themselves;
- (2) identifying the regions of the image to which these markings refer; and
- (3) reproducing those regions without interference from the markings themselves.

A method for finding word boxes or bounding boxes around image units is to close the image with a horizontal SE that joins characters but not words, followed by an operation that labels the bounding boxes of the connected image components (which in this case are words). The process can be greatly accelerated by using 1 or more threshold reductions (with threshold value 1), that have the effect both of reducing the image and of closing the spacing between the characters. The threshold reduction(s) are typically followed by closing with a small horizontal SE. The connected components labeling operation is also done at the reduced scale, and the results are scaled up to full size. The disadvantage of operating at reduced scale is that the word bounding boxes are only approximate; however, for many applications the accuracy is sufficient. The method described above works fairly well for arbitrary test fonts, but in extreme cases, such as large fixed width fonts that have large inter - character separation of small variable width fonts that have small inter - word separation, mistakes can occur. The most robust method chooses a SE for closing based on a measurement of specific image characteristics. This requires adding the following two steps:

- (1) Order the image components in the original or reduced (but not closed) image in line order, left to right and top to bottom.
- (2) Build a histogram of the horizontal inter - component spacing. This histogram should naturally divide into the small inter - character spacing and the larger inter - word spacings. Then use the valley between these peaks to determine the size of the SE to use for closing the image to merge characters but not join words.

A. Identifying the Hand Markings

Sub - problem (1) of the image interpretation problem involves identifying the markings. Several salient characteristics of the hand markings can be used to identify the markings. The characteristics include:

- (i) long horizontal, vertical, and oblique straight line segments, where "long" is relative to the size of machine marks, such as text characters;
- (ii) segments that are not exactly straight, having some curviness; and
- (iii) segments that are not horizontal or vertical, relative to the text in the image.

If the document consists only of text, without rules or line graphics, then it is not necessary to distinguish between hand markings and machine lines, and a probing of the image based on the length of the straight line segments is adequate to separate the hand markings from text. ("Probing" is typically done morphologically, optionally with reduction beforehand, using either an EROSION or an OPENING).

If the image may contain horizontal or vertical rules, it is necessary to distinguish the machine marks from the hand marks. In this case, the best results are obtained by utilizing all of the above characteristics. One method for distinguishing machine marks from hand marks are as follows:

- (a) Deskew the image as described in a copending patent application entitled "Method and Apparatus for Identification of Document Skew" to Bloomberg, et al. Serial No. 448,774, filed December 8, 1989, said copending patent application herein being incorporated in its entirety.
- (b) Do an OPENING of the image for long horizontal line segments. This will project out both machine - printed horizontal lines, and nearly horizontal hand-drawn line segments.

(c) For each connected component thus extracted, determine the width W , height H , and number N of ON pixels within a bounding box.

(d) Using the width, height and number of ON pixels within the bounding box, determine if the image segment is machine or hand made. This can be done by constructing factors such as: the ratio W/H (for horizontal segments); the ratio $N/(WH)$ (which designates a fractional area of ON pixels within the bounding box); the ratio $N/(W(H-c))$ (for horizontal segments), where c is a constant with workable values of about 2; the ratio $N/(H(W-c))$ (for vertical segments), where workable values for c are about 2; and comparing these with thresholds. If the constant c is 0, the special case occurs where the factor is $N/(WH)$. The reason for generalizing the factor $N/(WH)$ with the constant c is to compensate for jagged marks and slight misalignment on machine printed lines. For example, by removing 2 or so lines from the width, $N/(W(H-c))$ should be approximately 1.0 for machine printed lines, whereas it would be significantly smaller than 1.0 for handwritten marks.

B. Identifying the Marked Regions

Sub - problem (2) of the image interpretation problem provides for identifying regions of the document image to which the handwritten marks refer. The handwritten marks identified in sub - problem (1) are further processed to identify a target part of the document image.

A fill operation, starting with the identified segments as "seeds", and filling into the connected component of the image of which the identified segments are a part, will provide the connected hand marking. This marking can be an underline, a sideline, a circle, etc.

The asperity ratio (width to width) of the bounding box of the (filled) connected component can then be compared with thresholds to determine if the marks are underlines (large width/height), sidelines (large height/width), or circles (both width and height are larger than a minimum threshold value).

Underlines typically refer to the image units directly above them; image unit segmentation with association of the neighboring underline is appropriate. Thus, for example, the document image can be horizontally CLOSED (or DILATED) so that the letters within the image unit are merged; thus, when extracting by connected components, the entire image unit is obtained.

Sidelines typically refer to a block of the document image to the right of the sideline if the sideline is on the left side of the image, and vice versa.

Circles typically refer to a part of the document image that is encircled. Encircled means any marking that is intended to demarcate by enclosure, or near enclosure. The circle need not be closed, since the demarcated region, including the hand marking, may be determined by several methods as follows.

(a) Use a bounding box for the connected component. This is effective for isolated regions of the document image that are entirely circled, but it does not work well for circled marks that occur within a text block, for example, because unintended text within the bounding box but outside the contour will be lifted as well.

(b) Fill within the circled region. This is effective only when the region is CLOSED. There are several ways to test the connectedness of the region. A very simple method is to use a flood fill in either of two directions (filling from the inside or the outside). For a flood fill start with the extracted region of the image given by the bounding box of the connected component (perhaps using an image that has been slightly expanded beyond the bounding box, using OFF pixels outside the bounding box), either:

i) fill the background (OFF) pixels from the inside, and check if the fill escapes the circle, or

ii) fill the background pixels from the outside, and test if the fill penetrates the circle.

(c) Alternate direction CLOSING is effective for circles that are not closed contours. This method isolates the connected component and closes it along the smallest direction with a structuring element of size comparable to that dimension. Then close in the orthogonal direction with a structuring element of reasonable size (depending on the asperity of the connected component). Close again in the original direction and the result is a solid mask spanning the connected component. This can all also be done at reduced scale for efficiency.

C. Reproducing the Marked Regions

Sub - problem (3) of the image interpretation provides for reproducing demarcated regions without interference from the hand markings. Problems can arise when the hand marking is connected to some of the material that is to be extracted. Due to such connections, it is generally not possible to remove only the pixels produced by the hand marking from the extracted image. Nevertheless, solutions that work reasonably well do exist.

One method according to the invention is extracting those pixels that are believed to constitute the hand marking, and to XOR or set SUBTRACT them from the extracted image. If boundary noise pixels on the hand markings are not extracted, then these will remain in the image after the XOR or set SUBTRACTION; consequently, an OPENING with a structuring element needs to be used to remove them.

For sidelines there is usually no problem, since the sidelines are typically unconnected with the machine printed material. Underlines can touch the material above them. Consequently, an underline connected component should not be removed; since (for example) the connected component may include one or more characters of the text located above the underline. Instead, the underline should be extracted by horizontal OPENINGS alone without filling. The horizontal OPENING will not include boundary noise pixels. These noise pixels can be retrieved in one of two ways:

- (i) Either dilate the extracted underline using a small vertical structuring element. The DILATED underline will typically cover the boundary noise pixels, even when the structuring element is very small (say, 3 to 5 pixels high, with the center position in the center of the structuring element); or
- (ii) post-process the extracted (say, text) components by OPENING with a small structuring element to remove the noise.

As mentioned above, underlines are typically applied to image units that represent words, so the processing that is intended to lift the material above the underline should be word-oriented. After the image unit is extracted, the pixels representing the underline can be subtracted.

Circled regions are the most varied. Assume now that the demarcated region, including the hand markings, have been lifted out by the methods above. If the clipped region is an isolated part of the document image not touched by the circle, the entire region can be extracted, followed by XORing or SUBTRACTING the pixels of the circle component. If the hand marking touches machine printed text, the following options are available:

- (i) Remove the hand marking connected component, which would include all characters that touch it.
- (ii) Identify the hand marking pixels by a UNION of OPENINGS using structuring elements that represent line segments at a set of orientations (typically eight: 0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5 degrees). The size of the structuring elements is set by the criterion that they must be large enough to avoid including ordinary text characters, but small enough to get most of the hand marking. The UNION of OPENINGS image is then DILATED by a small isotropic structuring element (say, a 3x3 or 5x5 brick), and the result is XORed or SUBTRACTed from the lifted image.

To improve robustness of the above operations, the bitmap of each lifted region should be processed independently. Border pixels may need to be added to avoid boundary effects when CLOSING an image. By processing each region independently, the regions cannot interfere with each other (for example, a large CLOSING will not bridge two different regions). Further, the structuring elements chosen to process a region can be tailored to the dimensions and characteristics of each region, as in the above method for extracting regions demarcated by open circles.

III. Direct Approach

The methods and apparatus of the present invention disclose how regions of a document hand marked with an ordinary pen or pencil can be identified, even if marked through lines of text. Two characteristics of markup lines can be used in identifying and extracting those markup lines from a document image. First, the lines can be identified as being long lines and second, the markup lines can locally appear to be composed of straight line segments whereas text for example has local regions of high curvature.

A direct approach to the identification of markup lines, and subsequent processing of hand marked regions, extracts those parts of a document image that are composed locally of long straight lines, effectively removing the markup lines from the document image. The direct approach method is shown in a first preferred embodiment in Figure 1 where a first image of a document, as shown for example in Figure 5, is scanned in step 2. The first image of Figure 5 represents a scaled image at 150 pixels/inch. The first image is threshold reduced 2X in step 4 using a threshold of 2 to obtain a first image scaled at 75 pixels/inch. A morphological UNION of OPENINGS on the reduced first image of Figure 5 is taken in step 6, using a set of eight structuring elements as shown in Figures 27A-H, and forming a second image as shown in Figure 17. The structuring elements of Figure 27A-H each have a length 9 and width 1. They represent eight lines in different orientations. By taking a UNION of the OPENINGS, any pixels that are a part of the reduced first image that entirely covered by a structuring element pattern, are removed. The result in Figure 17 is quite clean with strong noise removed. The bounding boxes are created in step 8 and filled, forming a mask of the markup lines as shown in Figure 20.

Figure 2 shows a second preferred embodiment of an image markup method using the direct approach to the identification and extraction of hand marked regions. The first three steps of the method are identical to steps 2, 4 and 6 in Figure 1. Those first three steps include scanning a first image (from Figure 5) in step 2, reducing the image in step 4, and taking a UNION of OPENINGS of the first image using the set of eight structuring elements shown in Figures 27A - H in step 6. The departure from Figure 1 occurs in step 16 of Figure 2 where the second image shown in Figure 17 is reduced by 2X to 38 pixels/inch, using a threshold value of 1. The second image is processed by taking a UNION of the same set of OPENINGS used in step 6, i.e., the set of structuring elements shown in Figures 27A - H, to form a third image as shown in Figure 18. Since the second preferred embodiment occurs at a 2X reduction compared with the reduction in step 4 of the first embodiment, the structuring elements are effectively twice as long as they were in the prior step 4 UNION. As seen from Figure 18, the result of the morphological operations in providing the third image is a very clean image, effectively showing only markup lines. The third image is next reduced by 4X in step 22 by using two 2X reductions, each with threshold 1, creating a fourth image with a resolution of 9.5 pixels/inch as shown in Figure 19. A bounding box fill of the fourth image in step 24 results in a fifth image (mask) as shown in Figure 20. Small bounding boxes may be eliminated in step 26, providing error control for stray marks and extremely small enclosures that would not likely be purposely marked.

IV. Circle Extraction Technique

Another preferred embodiment for circle extraction is shown in the third embodiment of Figures 28 - 34 as described below, following the script text for the actual interpreted calls.

* TEXT SCRIPT

Copyright 1991 Xerox Corp.

```

* circle 7.scr
* 4x reduction
* Extraction by rabbf:
  intl = 1
* Extraction by edgefill (topology):
  intl = 0
moveToScale (1)
  * deskew
  pause(" deskew")
  deskew(prl, pr0, 4)
  copy(pw0, prl)

```

```

    * 4x reduction
    pause(" reduce 4x")
    reduceSelV(0, 1, 0, 0, 0, 0, 4)
5    pixrl = copyPixr(prl)
    copy(pw0, pr1)
    * extract hand drawn lines
10    pause(" get horizontal lines")
    dilate(pr2, pr1, stre12v)
    copy(pw0, pr2)
    open(prl, pr2, stre120h)
15    copy(pw0, prl)
    dilate(pr2, pr1, stre13)
    copy(pw0, pr2)
20    * identify all lines, and remove the short ones*/
    pause(" remove short horizontal lines")
    boxes11 = rabb(pr2)
    boxes12 = removeSmallBoxes(boxes11, 30, 2)
25    displayBoxes(pr3, boxes12)
    copy(pw0, pr3)
    * select hand written lines
    pause(" select hand written lines")
30    boxes13 = handLineDiscr(pr2, boxes12, 0.010)
    displayBoxes(pr3, boxes13)
    copy(pw0, pr3)
35    * use pixars to extract the hand marks at full scale
    pause(" extract circle at f.s.")
    pixar11 = makeEPixar(pr2, boxes13)
    pixar12 = expandPixar(pixar11, 4)
40    moveToScale(1)
    pixr12 = displayPixar(pixar12)
    and(pr2, pixr12, pr1)
    fillClipSeed(pr3, pr2, pr1)
45    copy(pw0, pr3)
    pause(" now get the contents")
    * use rabbf to get contents of bounding box
50    if (int1) {
        boxes14 = rabbf(pr3)

```

55

```

5      pixar21 = makeEPixar(pr3, boxes 14)
      pixar22 = makeEPixar(pr1, boxes 14)
      subtractPixar(pixar22, pixar22, pixar21)
      pixr22 = displayPixa
      r(pixar22)
      copy(pw0, pixr22)

```

10

The following script of C programming code has been implemented to determine whether marks on an image are handwritten or machine mode.

15

```

      * C CODE SCRIPT
      * Copyright 1991 Xerox Corp.
      /* handLineDiscr */

```

20

```

      #define MIN_ASPERITY_LIMIT 4.0 /* min for line */
      #define LOW_ASPERITY_LIMIT 25.0 /* handle differently */
      *define TOLERANCE_ANGLE 0.010 /* tolerance angle (in radians) allowed
      * for the machine line */
25      *define TOLERANCE_LINES 2 /* for low-asperity segments */

```

30

```

      /*****
      ***** Hand Line Discrimination *****
      *****/

```

35

```

      * Copyright 1991 Xerox Corp.
      /*

```

40

```

      * handLineDiscr(): takes a pixr, a boxes structure, and a tolerance
      * angle factor.

```

```

      * we assume the pixr has been deskewed, relative to some
      * text, that may (or may not) appear in the pixr.

```

```

      * The pixr may include only extracted horizontal or
      * vertical lines, or may have everything.

```

45

```

      * the boxes structure gives rects of either "horizontal"
      * or "vertical" lines, but not both.

```

```

      * if the input tolerance factor is 0.0, a default is
      * used.

```

50

```

      * returns a boxes struct giving locations of the hand
      * drawn lines, or NULL if there are no lines or on
      * error.

```

55

```

      */

```

```

5      #ifndef ANSI
        BOXES *
        handLineDiscr (pixr, boxes, tolerance)
        Pixrect      *pixr;
        BOXES      *boxes;
        double      tolerance;
10      #else
        BOXES * CDECL
        handLineDiscr
        (Pixrect *pixr,
15              BOXES *boxes,
                double tolerance)

        #endif /* ANSI */
20      {
        int      nRect, i, direction, w, h, numPix, area, excess;
        double   asperity, asperityHor, asperityVert, ratio;
        RN      *rlist, *rnode;
25      RECTANGL *rect;
        BOXES *boxesH;
        Pixrect *prSeg;
        static char procName[] = "handLineDiscr"
30

        if (!pixr)
            return abortP("source pr not defined", procName, NULL);
        if (!boxes)
            return abortP("boxes not defined", procName, NULL);
        rlist = NULL;
        if (tolerance == 0.0)
40            tolerance = TOLERANCE__ANGLE; /* use default */
            /* decide if boxes are horizontal or vertical:
                find the box with the largest asperity */
        Nrect = boxes->n;
45        asperityHor = 0.0;
        asperityVert = 0.0;
        for (i = 0; i < Nrect; i++)
50        {
            rect = boxes->rect[i];

```



```

    asperityHor = Max(asperityHor, ((double) rect->w / (double) rect->h));
    asperityVert = Max(asperityVert, ((double) rect->h / (double) rect->w));
5   }
    direction = (asperityHor > asperityVert) ? HORIZ : VERT;
    /* extract rects of hand lines */
    for (i = 0; i < Nrect; i + +)
10  {
        rect = boxes->rect[i];
        w = rect->w;
        h = rect->h;
15  if ((direction == HORIZ && h == 1) || (direction == VERT &&
            w == 1)) continue; /* straight and narrow: machine
                                made */
        if (direction == HORIZ)
            asperity = (double) w / (double) h;
        else /* vertical */
            asperity = (double) h / (double) w;
25  if (asperity < MIN__ASPERITY_LIMIT) /* not a line segment */
            continue;
        if ((prseg = rectOfPixrect(pixr, rect)) == NULL)
            return abortP("prseg not made", procName, NULL);
30  numPix = numberPr(prseg, PIXELS);
        area = w * h;
        ratio = (double) numPix / (double) area;
        if (direction == HORIZ) {
            if (asperity < LOW__ASPERITY_LIMIT)
                excess = w * (h - TOLERANCE__LINES) - numPix;
            else
40  excess = w * (h - tolerance * w) - numPix;
        }
        else { /* vertical */
            if (asperity < LOW__ASPERITY_LIMIT)
55  excess = h * (w - TOLERANCE__LINES) - numPix;
            else
                excess = h * (w - tolerance * h) - numPix;
        }
        if (excess > 0) /* hand drawn; add rect to list */

```

```

    {
        if ((rnode = createRectNode(rect)) == NULL)
            return abortP("rnode not made", procName, NULL);
        rlist - addToHeadRL(rlist, rnode);
    }
    destroyPixr(prseg);
}
if (!rlist)
    return abortP("no hand written lines identified", procName, NULL);
if ((boxesH = RLToBoxes(rlist)) == NULL)
    return abortP("boxesH not made", procName, NULL);
/* clean up */
destroyRL(rlist);

```

V. Preferred Method for Circle Extraction

Figure 28 deskews the original image shown. The deskew procedure is optional, but it helps when using the handLineDiscrimination procedure. A 4X reduction of the first image of Figure 28 is provided and shown in Fig. 29 (although a 2X reduction is also acceptable). The reduction procedure is optional, but it helps efficiency, since many of the subsequent operations can be performed much faster on a reduced image. In Figure 30, horizontal lines have been extracted with an OPENING (or less satisfactorily an EROSION) using a long horizontal structuring element. The structuring element used in this case was 20 pixels long, which is reasonable for images with resolution of 40 - 80 pixels/inch which corresponds to a reduction between 4X and 8X from typical scanning resolution of 300 pixels/inch. If the hand line is weak, i.e., thin, and slanted, it may not be possible to extract a long horizontal line. To handle thin lines, a small vertical dilation can be done before the horizontal OPENING to make the lines thicker. Furthermore, another DILATION can be done after horizontal line extraction to join slightly separated components. In Figure 31, the connected components have been found and short lines have been removed using any number of conventional techniques. Figure 31 shows bounding boxes for those components remaining after the short lines have been culled. In Figure 32, handwritten lines are selected. Possible discriminating factors (both used in this example) include the asperity ratio (width to height) and the fraction of the bounding box containing ON pixels. In Figure 33, the handwritten marks are extracted at full resolution as follows.

- (1) Expand the segments selected in Figure 32 to full resolution.
- (2) AND expanded segments from Fig. 32 with the original image of Figure 28.
- (3) Use the result of the ANDing operation as a seed in a filling operation, with the original image of Figure 28 as the clipping mask, to generate the entirety of the original hand marks.

The contents of regions delineated by the hand marks (as shown in Figure 34) have been obtained by (1) determining the bounding boxes of the connected components, (2) extracting the bounding box regions from the original image, and (3) SUBTRACTING the hand marks (shown in Figure 33). This method can be used even if the hand marks do not formed a closed curve.

A general approach to the method of extracting encircled regions of a document image which have been hand marked with an ordinary pen or pencil (excluding a highlighter pen) is illustrated in the flowchart of Figure 46. An image is scanned in step 200 and the resultant document image A is deskewed in step 202. The deskewed image B is then OPENed in step 204 using at least one horizontal or one vertical structuring element. The OPENed image C then has either horizontal or vertical image segments according to the type of structuring element used in step 204. Bounding boxes are determined about image units in step 206 resulting in a data structure D which includes bounding rectangles of horizontal or vertical line segments. In step 208, bounding boxes are removed from the data structure D which include only machine markings, resulting in a data structure E. The regions represented by the data structure E in step 210 are extracted from the deskewed image B, resulting in an image F of hand drawn line segments. The hand

drawn line segments of image F are used as a seed in step 212 for a seed fill operation using image B as a clipping mask. The result of the seed fill operation is an image G of filled circles representing the hand drawn marks. In step 214, bounding boxes are found around the filled circles of image G, resulting in a data structure H which includes bounding rectangles of hand drawn marks. The regions represented by the data structure H are then extracted from the deskewed image B in step 216, resulting in an image J containing hand drawn marks and contents. In step 218, image J (containing hand drawn marks and contents) is exclusive-ORed with image G. The result of the XOR is an image containing only contents of hand drawn regions, i.e., no hand drawn marks remain in the image.

10 IV. Indirect Approach

An indirect approach to the identification of hand markup lines involves removing all print from the document image except the markup lines. The indirect approach shown by the steps in Figure 3 begins in step 30 by scanning a first image, shown in Figure 5, at 150 pixel per inch. The scanned first image of Figure 5 can be created, for example, by scanning at 150 pixels/inch or by scanning at 300 pixels/inch then reducing by a factor of 2 with a threshold of 2. The second image shown in Figure 6 is formed by CLOSING the first image of Figure 5 with a first horizontal structuring element such as the structuring element of length 8 shown in Figure 21. Other structuring element orientations can also be used, but the horizontal orientation is most effective in CLOSING up curves and spaces between characters. The CLOSING operation in step 32 identifies regions of high curvature in the first image. The result of the CLOSING operation of step 32 (Figure 6) and the first image of step 30 (Figure 5), are XORed at the XOR gate 34 shown in Figure 3. The result of the XOR is shown in Figure 7 as a third image which includes only those pixels that were added by the CLOSING operation of step 32. Most of those pixels are expected to be near lines that have high curvature since lines with low curvature should not generate many such pixels. A fillClip operation on the third image of Figure 7 is segmented into two steps in Figure 3, namely DILATING the third image in block 36 followed by logically ANDing the result of the DILATION with the first image in step 38. Specifically, the third image of Figure 7 is DILATED in step 36 with a solid 3 x 3 structuring element. The DILATED third image is then ANDed with the first image in step 38 forming the fourth image of Figure 8. The fillClip process fills back the lines of the document that generated the high curvature pixels in Figure 6. In order to completely fill back the lines of the document as shown in Figure 8, the fillClip operation can be iterated more than once. Alternatively, the third image of Figure 7 could be sequentially DILATED with a solid 3 x 3 structuring element, thence logically ANDing the result of the DILATION with the first image of Figure 5 to produce a fourth image as shown in Figure 8. The fourth image of Figure 8, output from AND gate 38, is then XORed in gate 40 with the first image of Figure 5 from step 30, forming a sixth image as shown in Figure 10. In order to strengthen the markup lines and eliminate background noise, many alternative morphological processes can be used to obtain the fifth image shown in Figure 9. The method chosen in this embodiment includes taking an UNION in step 42 of the fifth image of Figure 9 with a set of openings, specifically using a set of four structuring elements as shown in Figures 23-26. The morphological operation involves OPENING the fifth image of Figure 9 by each of the structuring elements in turn, then taking a UNION of the results, forming a sixth image shown in Figure 10 with reduced noise, observed as containing generally thin vertical lines, wherein every pixel in Figure 10 belongs to a run of 5 pixels in at least one of the four orientations of the structuring elements shown in Figure 23-26. The background noise is further reduced by OPENING the sixth image of Figure 10 in step 44 with a second horizontal structuring element of length 2 as shown in Figure 22, forming the seventh image of Figure 11. At this point, the background noise is sufficiently reduced to allow a number of alternatives for continued image processing, such as CLOSING the seventh image with a sequence of structuring elements to close up the small gaps in the markup lines, followed by small openings to remove the background noise. In step 46 the seventh image is reduced by a factor of 4, i.e., from 150 pixels/inch to 38 pixels/inch, using two threshold reductions of 2X with threshold 1 for each reduction, producing an eighth image as shown in Figure 12. Next, the eighth image of Figure 12 is CLOSED in step 48 in sequence with the fourth structuring elements shown in Figures 23-26, forming a ninth image as shown in Figure 13. Some of the breaks in the curves of the eighth image of Figure 12 have now been closed in the ninth image of Figure 13. In step 50, a UNION of the ninth image with the OPENINGS of the same set of four structuring elements shown in Figures 23-26 results in a tenth image free from most background noise as shown in Figure 14. The tenth image of Figure 14 is reduced in step 52 by a factor of 2, forming an eleventh image as shown in Figure 15. A bounding box fill is performed on the eleventh image of Figure 15 in step 54, forming the twelfth image of Figure 16. The twelfth image in Figure 16 depicts a bounding box mask which is expanded as shown in step 56 to full scale, then logically ANDed with the first image (of Figure 5) from step 30, in AND gate 58.

wherein the process of image markup detection is completed at step 60.

VI. Topological Methods

5 Another preferred embodiment of a method for extraction of the contents of a closed circled region includes extracting handwritten marks at full resolution as shown in Figure 35 (which is the same result as Figure 33 of the previously described embodiment). In Figure 36, the bounding boxes of the connected components have been obtained. For efficiency and to prevent interaction between different parts, each subimage is extracted and handled separately. Figures 37-42 show results at intermediate stages of
10 processing, where the subimages have been put together in their original location, forming a composite image. In Figure 37, each subimage has been edge-filled (in a first edge-fill) by doing a flood fill from the edges for each subimage. If the hand mark is closed, the fill will not penetrate the interior. Note that two of the hand "circles" are opened, one is closed, and the graphics subimage is acting like a closed circle. In Figure 38, each subimage is inverted (in a first, edge-fill invert). If the hand mark contour is open, there will
15 be little left (just the holes of text, for example). If the circle is closed, the contents appear as white on a black background. Note that the previous two steps can be used to decide whether a closed contour exists by testing for the number of ON pixels compared to the area of contour. If the ON pixels occupy a significant fraction of the area (for example, more than 30%), then the contour is closed. Alternatively, if after a small EROSION there are no ON pixels left, the contour is open.

20 The contents of the closed contour can be extracted in several ways. For example, each inverted subimage can be edge-filled (in a second edge-fill) as shown in Figure 39. A solid subimage of ON pixels results for open contours, whereas the edge-fill just fills the exterior part of the closed contours. Figure 40 shows each subimage inverted (in a second edge-fill invert) whereby the interior is extracted. The two open contours have no interior, whereas the closed contour subimages yield the contents shown. Since the
25 graphic subimage appeared as a closed contour, its non-zero contents also are displayed. The basic operation of content extraction is edge-fill/invert, which is a topological operation extracting the interior ON pixels within a closed contour, noting that the shape of the various image components is not important.

Another topological method for extracting the contents of closed circled region (shown in Figure 44), includes: scanning a document image in step 250; filling the document images from the edges in step 252
30 (which fills the whole image except enclosed regions); bitwise inverting the image in step 254 so that only the enclosed regions are filled; filling from the edges in step 256; and bitwise inverting the image in step 258, resulting in an image with only hand drawn circled regions filled, the first edge-fill operation is provided to exclude machine made encircled regions.

A semi-topological method for extracting the contents of a closed circled region is shown in Fig. 45.
35 The method includes: scanning a document image in step 280; filling the document image from the edges in step 282; bitwise inverting the image in step 284; OPENING the image using a solid structuring element in step 286; CLOSING the image using at least one vertical structuring element in step 288; and logically ANDing the CLOSED image with the original image scanned in step 280, resulting in an image which includes only those regions which have been hand drawn as circles.

40 The following script was used for producing the images shown in Figures 36-40 when starting with the image shown in Figure 35.

45

50

55

*TEXT SCRIPT

* Copyright 1991 Xerox Corp.

- * use topological method to get contents of the hand marks;
- * this only works if the marks form closed curves

```
pause ("get contents: first fill/invert...")
```

```
boxes14 = rabbf(pr3)
```

```
* displayBoxes(pr2, boxes14)
```

```
* copy(pw0, pr2)
```

```
pixar21 = makeEPixar(pr1, boxes14)
```

```
pixar22 = edgeFillPixar(pixar21)
```

```
* pixar22 = displayPixar(pixar 22)
```

```
* copy(pw0, pixr22)
```

```
invertPixar(pixar22)
```

```
pixr22 = displayPixar(pixar22)
```

```
copy(pw0, pixr22)
```

```
pause (" and fill/invert again")
```

```
pixar23 = edgeFillPixar(pixar22)
```

```
* pixr23 = displayPixar(pixar23)
```

```
* copy(pw0, pixr23)
```

```
invertPixar(pixar23)
```

```
pixr23 = displayPixar(pixar23)
```

```
copy(pw0, pixr23)
```

As an alternative to the second edge-fill/invert procedures as described above as shown in Figs. 39 and 40, is as follows. After the first edge-fill/invert, there are noise pixels from inner loops of characters remaining in the regions with open circles. These can be eliminated using an erosion or, preferably, an opening. Figure 41 shows the result after an opening. Selecting only the non-images; i.e., those whose contours are closed (as described in the first edge-fill invert above) the result, in Figure 41, is that there are now only two subimages. Then each subimage can be closed with a SE large enough to convert the background text to ON pixels, so that the resulting bitmap is a region of solid ON pixels. After closing using a SE that corresponds to 0.13 inch in the original image (the size can vary), the result in Fig. 42 is obtained in two ways:

The procedure follows:

(a) AND this "mask" with the original image, to extract the text. If this is done, a few pixels from the hand-drawn contour will typically also be included. The reason is that the closing operation will grab a few of the innermost pixels of this boundary.

(b) Before ANDing the "mask" with two original image, first erode it with a small SE (say, 3x3). This should prevent it from overlapping with the hand-drawn contour; thus, (b) is a better method than (a). The result is shown in Fig. 43.

The following script file created the images shown in Figs. 36-38, and Figs. 41-43 (the alternate method, described above.)

* SEMI-TOPOLOGICAL

* Copyright 1991 Xerox Corp.

* use semi-topological method to get contents of the hand marks;

* this only works if the marks form closed curves

pause (" get contents: first fill/invert...")

boxes 14 = rabbf(pr3)

* displayBoxes(pr2, boxes14)

* copy(pw0, pr2)

pixar21 = makeEPixar(pr1, boxes14)

pixar22 = edgeFillPixar(pixar21)

* pixr 22 = displayPixar(pixar22)

* copy(pw0, pixr22)

invertPixar(pixar22)

pixr22 = displayPixar(pixar22)

copy(pw0, pixr22)

pause ("open, close, erode, and with the original")

openPixar(pixar22, pixar22, strel6h, strel6v)

pixr22 = displayPixar(pixar22)

copy(pw0, pixr22)

pixar23 = addBorderPixar(pixar22, 10, 0)

closePixar(pixar23, pixar23, strel20h, strel20v)

pixr23 = displayPixar(pixar23)

copy(pw0, pixr23)

erodePixar(pixar23, pixar23, strel3h, strel3v)

pixr23 = displayPixar(pixar23)

copy(pw0, pixr23)

pixar24 = removeBorderPixar(pixar23, 10)

andPixar(pixar24, pixar21)

pixr24 = displayPixar(pixar24)

copy(pw0, pixr24)

The addBorderPixar and removeBorderPixar operations are a small implementation detail to prevent problems at the image boundaries in the CLOSING operation. Since there is always a border of white space, there are no image boundary issues.

VII. Image Hand Mark Detection Device

The image hand mark detection device 80 of Fig. 4 includes, for example, a processor 84, monitor 86, read only memory 88, random access memory 90, files 92 and an output device 94. Device 80 represents
 5 any number of automatic devices for altering a document image such as a computer system, a facsimile system or a photocopying system. The user input device 80 represents any user input device such as a keyboard, optical scanner electronic scanner, photocopier or facsimile machine. Once a hand marked document image (not shown) is read by the user input device 82, the document image is processed by processor 84 to extract the hand marks or the contents desired in accordance with the methods described
 10 herein. The processor 84 operates by way of instructions provided by ROM 88 and information provided by RAM 90 whereby access to files 92 can be obtained. The results of the extraction of the hand marked document image segments can be displayed in monitor 86 or output to output device 94 (which represents any output device for outputting a document image such as a facsimile machine, photocopier, printer or CRT display).

15 The previously described apparatus and methods have shown how to identify and extract markup lines from a source image, using characteristic features of both the markup lines and the other print in the source image. The current invention has used only morphological operations and threshold reductions wherein all morphological operations are translationally invariant and can be performed with both parallel and pipelined architectures, thus affording extremely fast and cost effective implementations.

20 While the present invention has been described with reference to particular preferred embodiments, the invention is not limited to the specific examples given, and other embodiments and modifications can be made by those skilled in the art without departing from the scope of the invention as defined in the following claims.

25 Claims

1. A method for processing a scanned first image in a digital computer for differentiating machine marks from hand marks and identifying a location of non-transparent hand marks and hand marked regions, the method comprising the steps of:
 30 identifying and differentiating markings on the scanned first image as hand and machine marks using characteristics of the markings, said characteristics including horizontal, vertical, oblique, curved and irregular shapes;
 identifying regions of the scanned first image to which the hand marks refer; and
 reproducing the identified regions of the scanned first image to which the hand marks refer without
 35 interference from the hand marks.
2. A method for processing a scanned first image in a digital computer to identify a location of non-transparent hand marks and hand marked regions, characterised by
 (a) taking a union of OPENINGS using a first set of structuring elements with the first image forming
 40 a second image,
 (b) creating bounding boxes around the mark-up lines of the second image, said bounding boxes identifying the mark regions.
3. A method as claimed in claim 2, characterised in that step (a) further comprises reducing the first
 45 image.
4. A method for processing a scanned first image in a digital computer to identify a location of non-transparent hand marks and marked regions, characterised by
 (a) CLOSING the first image with a first horizontal structuring element, forming a second image;
 50 (b) XORing the first image with the second image, forming a third image;
 (c) DILATING the third image with a solid square structural element, ANDing the DILATION of the third image with the first image, forming a fourth image;
 (d) XORing the fourth image with the first image, forming a fifth image;
 (e) taking a union of multiple OPENINGS of a first set of structuring elements and the fifth image,
 55 forming a sixth image;
 (f) OPENING the sixth image with a solid square structuring element, forming a seventh image;
 (g) reducing the seventh image by a first reduction factor, forming an eighth image;
 (h) CLOSING the eighth image with the first set of structuring elements, forming a ninth image.

- (i) taking a union of OPENINGS of the ninth image with the first set of structuring elements, forming a tenth image;
 - (j) reducing the tenth image by a second predetermined factor, forming an eleventh image;
 - (k) filling the bounding boxes of the eleventh image, forming a twelfth image;
 - (l) expanding the twelfth image to a full scale; and
 - (m) ANDing the twelfth image with the first image, extrapolating the hand marked region.
5. A topological method for processing an image in a digital computer for extraction of regions of a document image encircled by non-transparent hand marks, characterised by
 - (a) flood filling the document image from edges;
 - (b) bitwise inverting the document image;
 - (c) flood filling the document image from edges; and
 - (d) bitwise inverting the document image.
6. A semitopological method for processing an image in a digital computer for extracting regions of a document image encircled by non-transparent hand marks, characterised by
 - (a) flood filling the document image from edges;
 - (b) bitwise inverting the document image;
 - (c) OPENING the document image using a solid structuring element of a first predetermined size;
 - (d) CLOSING the document image using a solid structuring element of a second predetermined size; and
 - (e) ANDing the CLOSED document image with the original document image.
7. A method for processing a document image in a digital computer for identification in the document image of hand drawn lines comprising the steps of:
 - (a) OPENING the document image using at least one structuring element;
 - (b) finding bounding boxes around image units in the document image; and
 - (c) testing the document image to identify horizontal hand drawn lines.
8. A method for processing a document in a digital computer for identifying hand drawn encircled regions of the document image, the method comprising the steps of:
 - (a) OPENING the document image;
 - (b) finding bounding boxes of image units in the document image;
 - (c) testing the document image to identify hand drawn lines; and
 - (d) filling into the original document image, using a result of OPENING the document image as a seed and filling into the original image.
9. A method as claimed in claim 7, characterised by
 - (a) filling into the original document image, using a result of OPENING the document image as a seed and filling into the original image and
 - (b) XORing a result of step (d) with the original document image so as to remove hand drawn marks from the document.
10. An apparatus for processing a scanned first image in a digital computer for differentiating machine marks from hand marks and identifying a location of non-transparent hand marks and hand marked regions, characterised by
 - image scanning means for scanning a first image;
 - processor means for identifying and differentiating markings on the scanned first image as hand and machine marks using characteristics of the markings, said characteristics including horizontal, vertical, oblique, curved and irregular shapes;
 - identification means for identifying regions of the scanned first image to which the machine marks refer;
 - reproducing means for reproducing the identified regions of the scanned first image so that the hand marks refer without interference from the machine marks; and
 - an output means for outputting the reproduction of the identified regions.

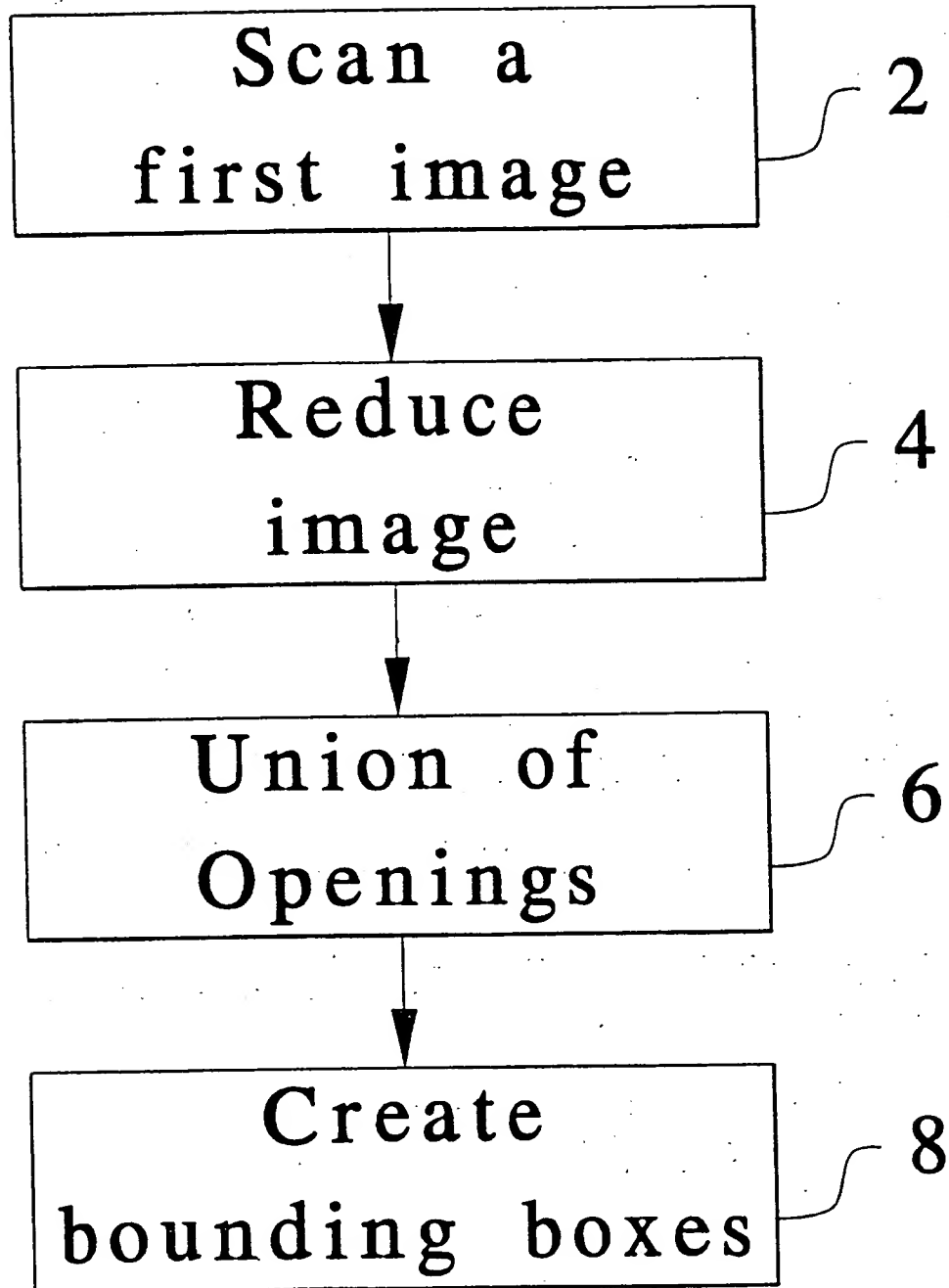


Fig. 1

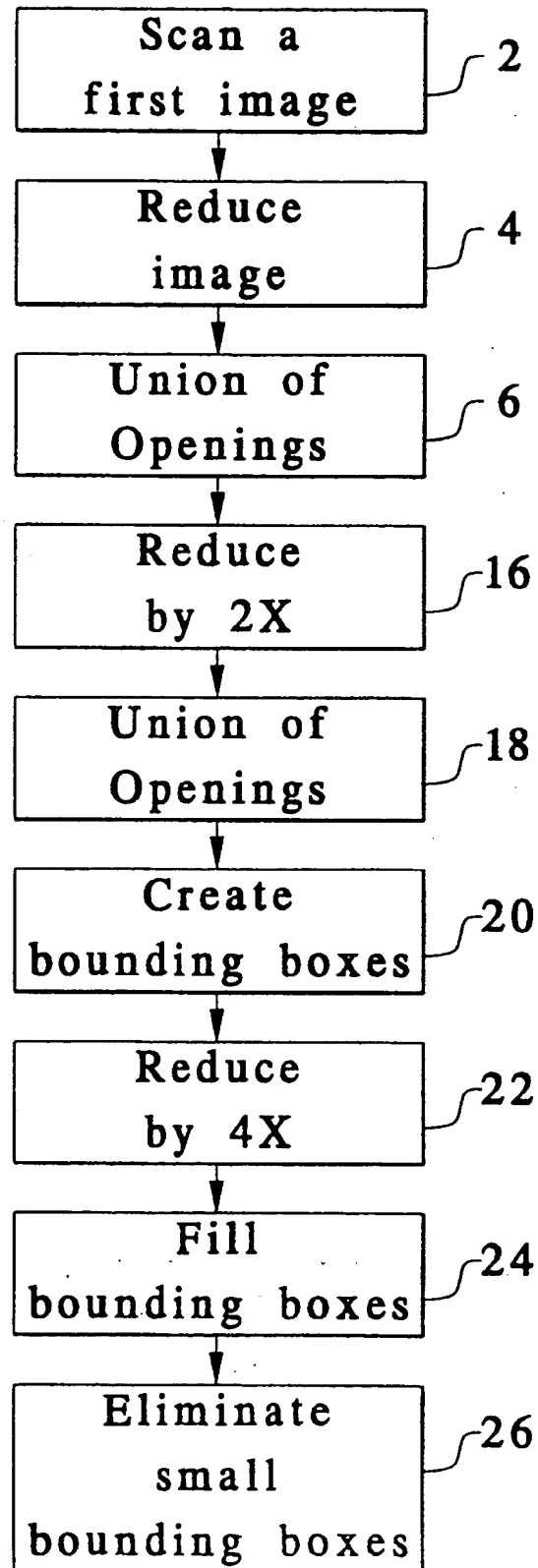


Fig. 2

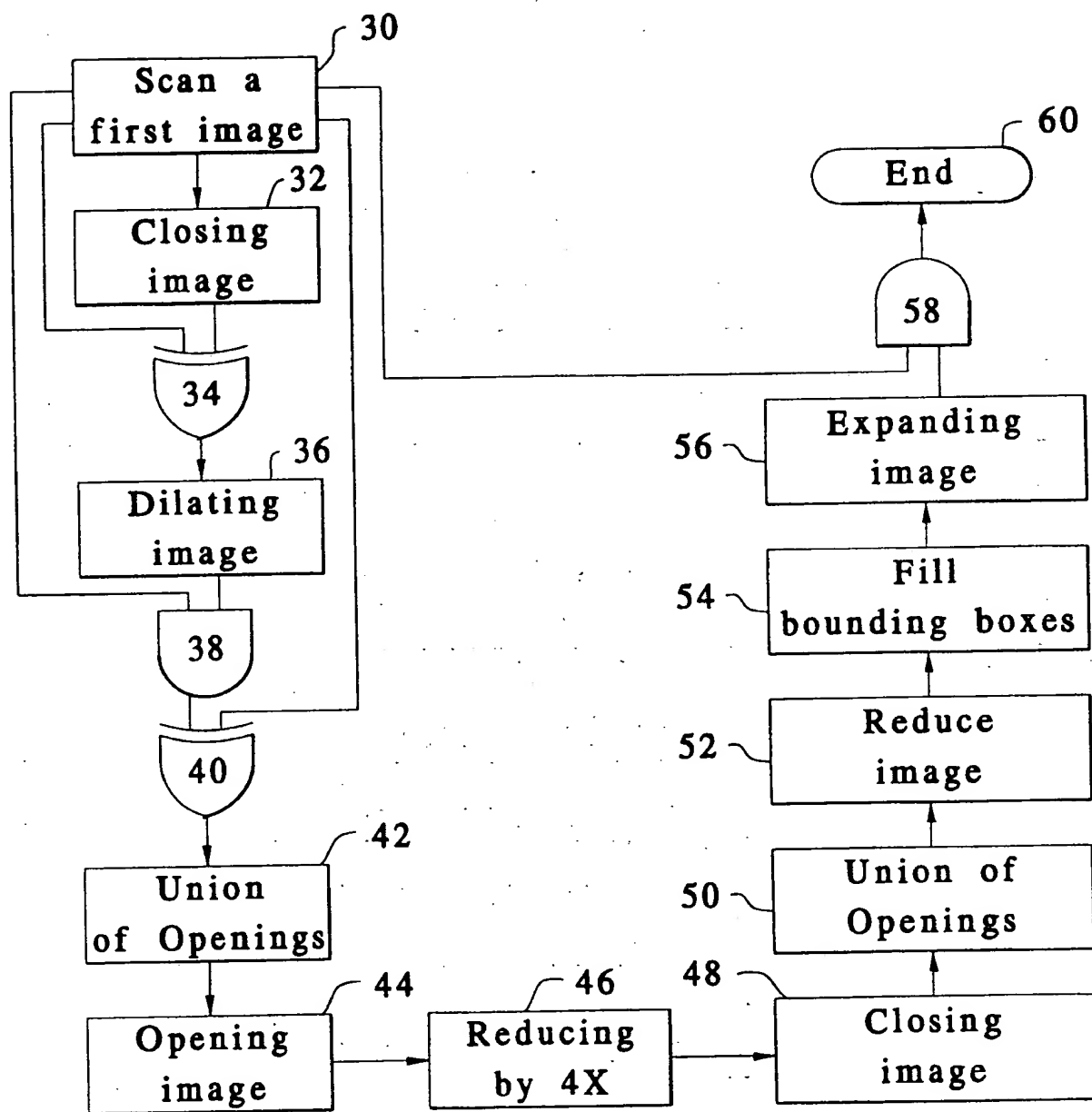


Fig. 3

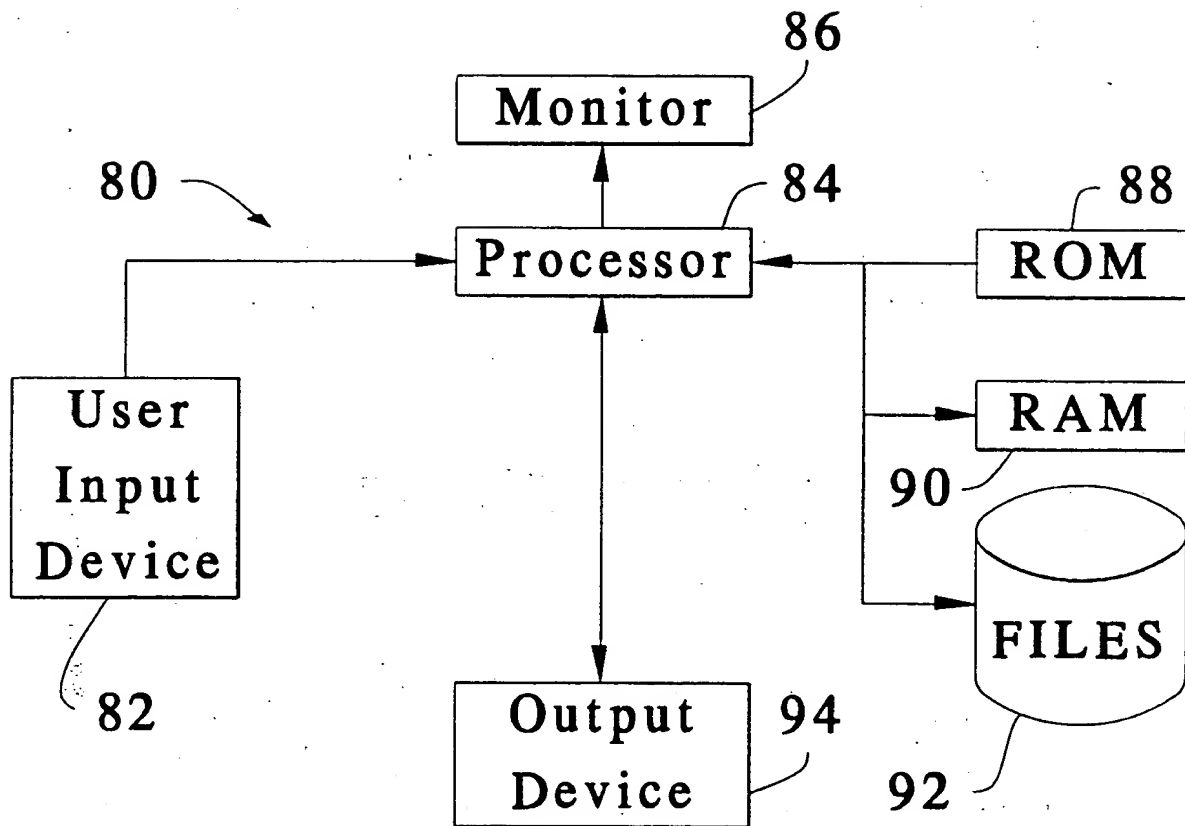


Fig. 4

Initially, the monocomponent developers which were introduced depended on toners which were conductive; however, this process has largely been replaced by resistive toner. The conductive toner could be charged easily in the nip by induction but has a severe drawback when transfer to plain paper is considered. The toner charge could be easily lost because of its humidity sensitivity. In addition, the transfer efficiency is lowered due to the moisture in the paper. Resistive toner, introduced by 3M in 1977 (Field, 1981), has the advantage that the toner behaves as if it were conductive in the nip for charging but after development its resistivity is high enough for transfer to plain paper.

Another variation of monocomponent development uses a toner with no magnetic additives. This system uses a donor roll covered with a material that will charge and transport the toner to the photoconductor. Although there are problems with deposition and charging of a uniform layer of toner on the donor roll, high quality prints have been demonstrated (Chang and Wilbur, 1974). Nonmagnetic systems create difficulties in toner manipulation but also make possible the lowering of toner cost, and the use of toners in other than dark colors. Ricoh is using their version of this development method in a recently announced copier machine utilizing colored toner (Repro-Jr.).

Liquid development permits, in principal, the use of the simplest hardware and offers the possibility for the highest resolution print due to the small size of its toner particles. This type of development uses a hydrocarbon dispersion of very fine toner particles that have been charged by the natural electrical double layer that arises during solvation. This mode of development has the advantages of simplicity, low cost, and does not require thermal fusing because "fixatives" can be added to the ink. There are safety and environmental concerns, however, arising from the use of liquid chemicals in an office environment which have limited the use of this technology to date.

Fig. 5

Initially, the nondevelopmental development which was introduced developed on water which was contained between two plates and largely been replaced by various types. The continuous water could be changed only in the top by immersion but has a static drawback when transfer to plain paper is considered. The water change could be only last because of its increasing viscosity. In addition, the transfer efficiency is limited due to the movement in the paper. Another water, introduced by Jd in 1977 (Field, 1977), has the advantage that the water behaves as if it were conducting a strongly changing two-phase development as viscosity is high enough for transfer to plain paper.

Another variation of nondevelopmental development was a water with no suspending addition. This system was a donor cell coated with a material that will change and transport the water to the photoconductor. Although there are problems with deposition and changing of a uniform layer of water on the donor cell, high quality prints have been demonstrated (Cheng and Wilson, 1979). Nondevelopmental development is water transportation but also makes possible the lowering of water cost, and the use of water in other than dark colors. Even is using the variation of this development method in a recently announced capillary machine utilizing contact water (Kopco, Inc.).

Liquid development systems, in principle, the use of the developer itself, were first offered the possibility for the highest resolution print due to the small size of its micro-particles. This type of development was a liquid-based dispersion of very fine water particles that have been changed by the natural chemical developer layer that occurs during exposure. This mode of development has the advantages of simplicity, low cost, and does not require chemical fixing because "fixatives" can be added to the ink. There are safety and environmental concerns, however, arising from the use of liquid chemicals in an office environment which have limited the use of this technology to date.

Fig. 6

[illegible][illegible][illegible]

Fig. 7

Initially, the monocomponent developers which were introduced depended on toners which were conductive; however, this process has largely been replaced by resistive toner. The conductive toner could be charged easily in the nip by induction but has a severe drawback when transfer to plain paper is considered. The toner charge could be easily lost because of its humidity sensitivity. In addition, the transfer efficiency is lowered due to the moisture in the paper. Resistive toner, introduced by JM in 1977 (Field, 1981), has the advantage that the toner behaves as if it were conductive, ~~in the nip for charging but after development~~ its resistivity is high enough for transfer to plain paper.

Another variation of monocomponent development uses a toner with no magnetic additives. This system uses a donor roll covered with a material that will charge and transport the toner to the photoconductor. Although there are problems with deposition and charging of a uniform layer of toner on the donor roll, high quality prints have been demonstrated (Chang and Wilbur, 1974). Nonmagnetic systems create difficulties in toner manipulation but also make possible the lowering of toner cost, and the use of toners in other than dark colors. Ricoh is using their version of this development method in a recently announced copier machine utilizing colored toner (Repro-Jr.).

Liquid development permits, in principle, the use of the simplest hardware and offers the possibility for the highest resolution prints due to the small size of its toner particles. This type of development uses a hydrocarbon dispersion of very fine toner particles that have been charged by the natural electrical double layer ~~that forms~~ during solvation. This mode of development has the advantages of simplicity, low cost, and does not require thermal fusing because "fixatives" can be added to the ink. There are safety and environmental concerns, however, arising from the use of liquid chemicals in an office environment which have limited the use of this technology to date.



Fig. 8

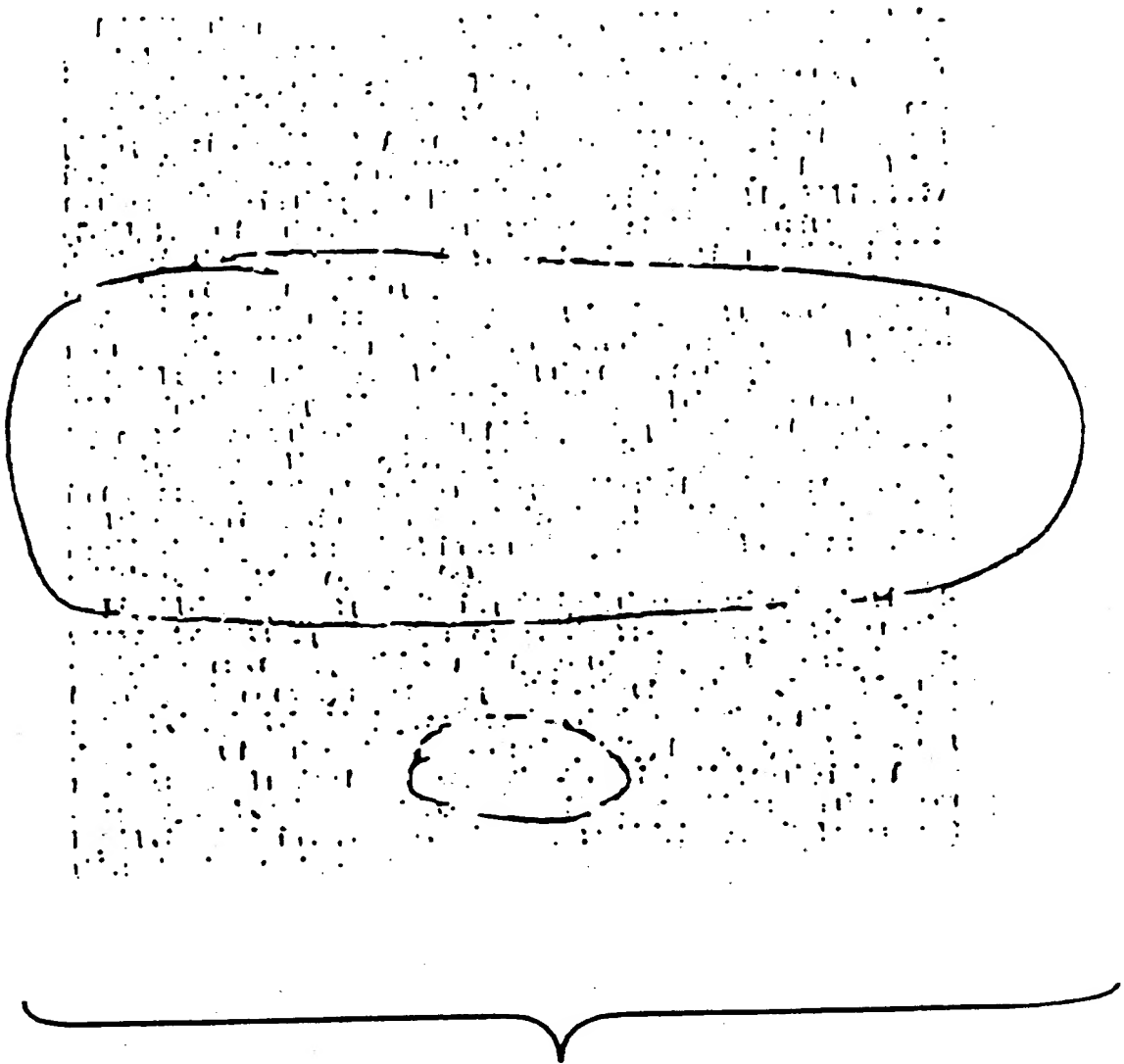


Fig. 9

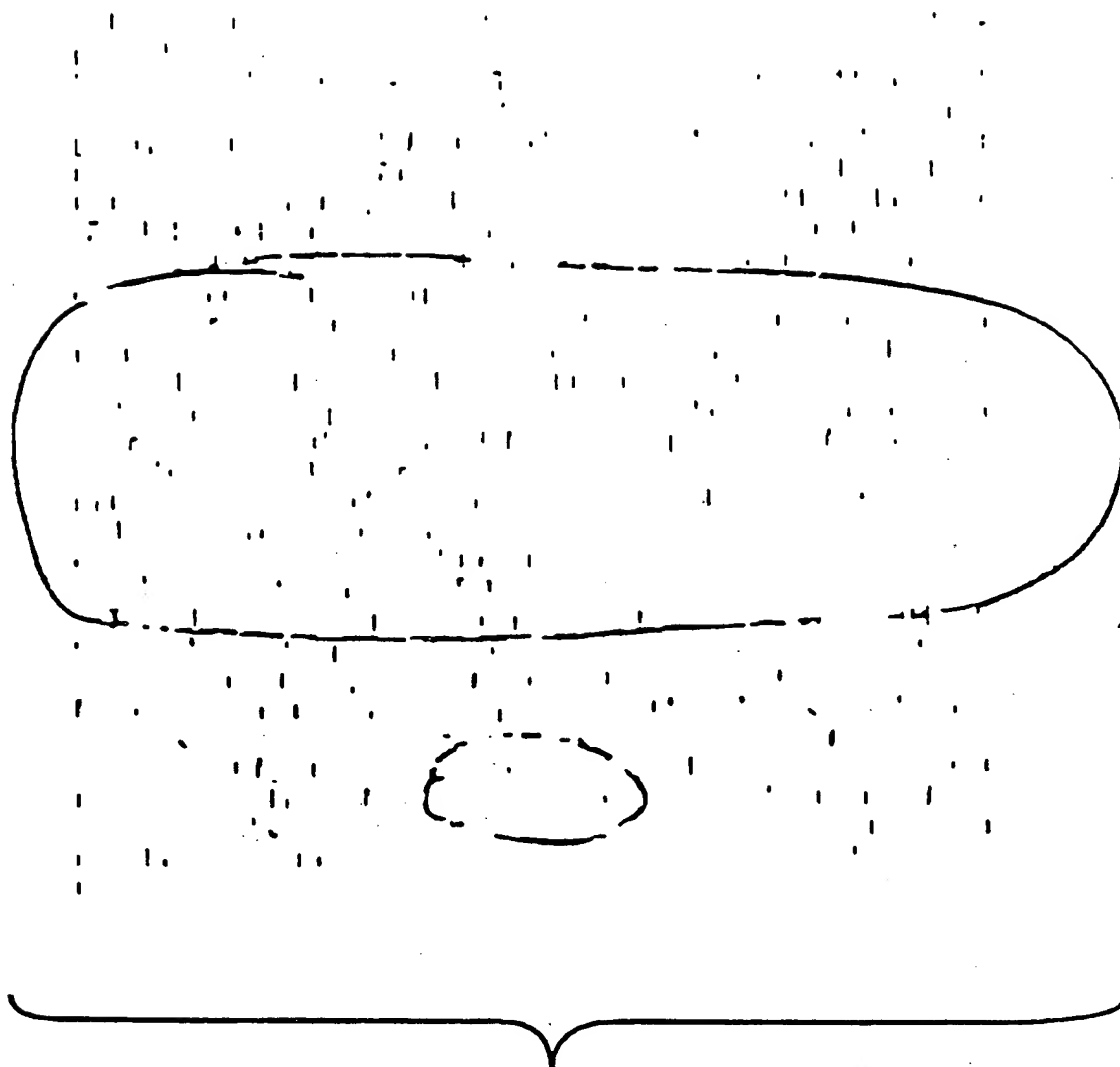


Fig. 10

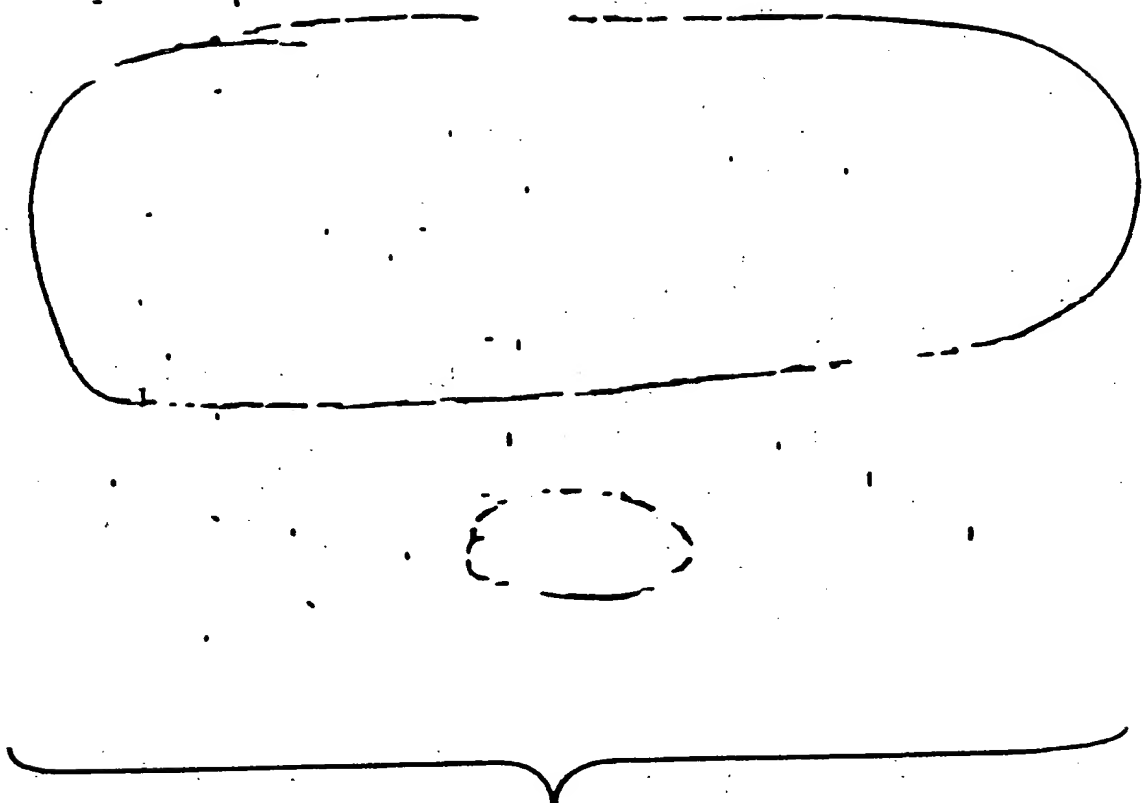


Fig. 11

SA EP.0 543 599 A2

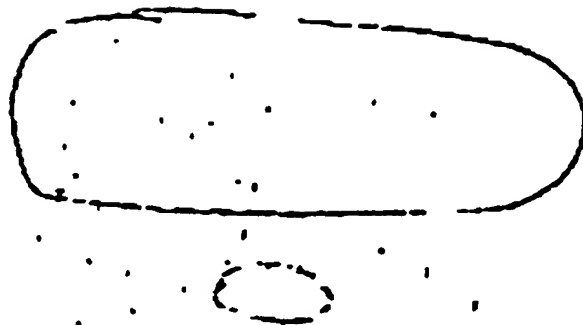
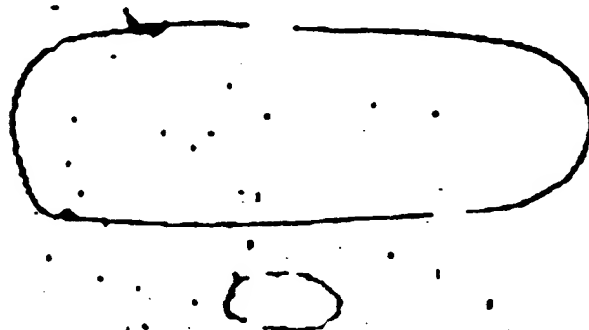


Fig. 12

Fig. 13



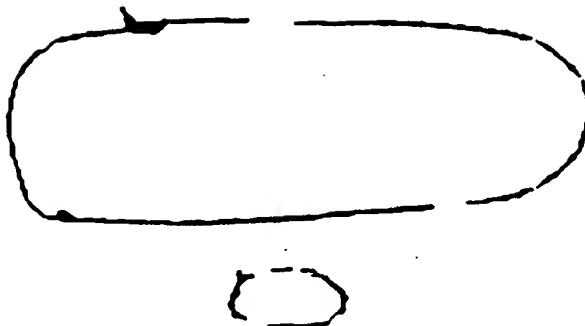
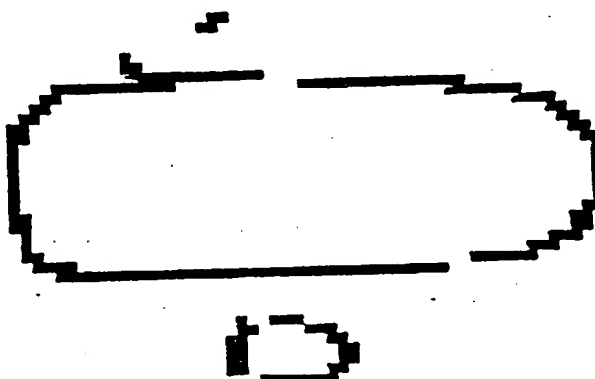


Fig. 14

Fig. 15



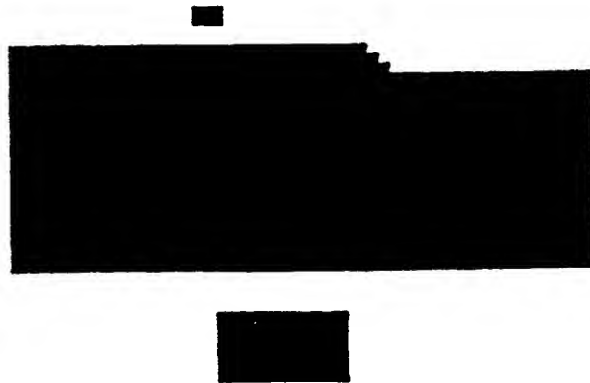
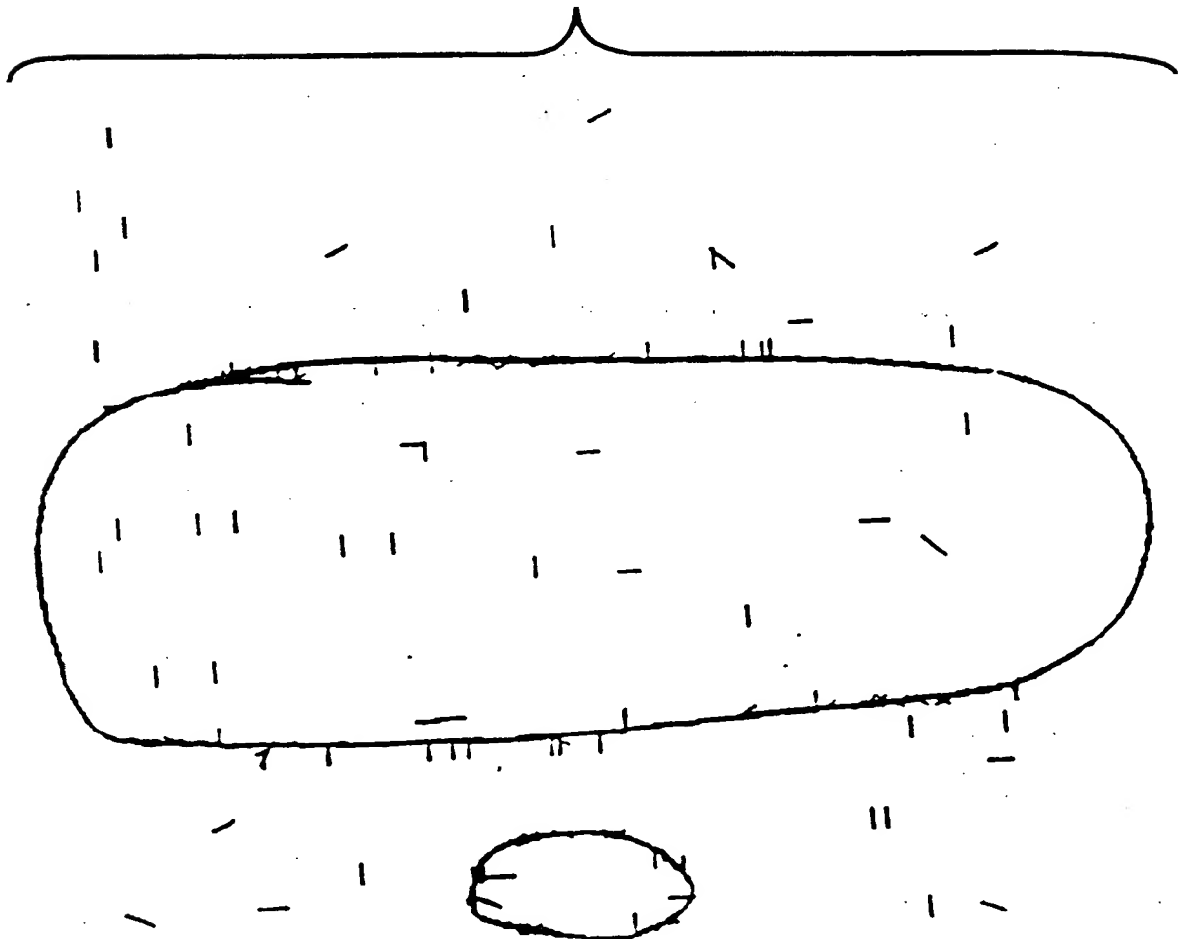


Fig. 16
Fig. 17



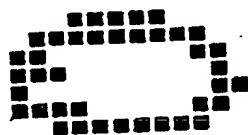
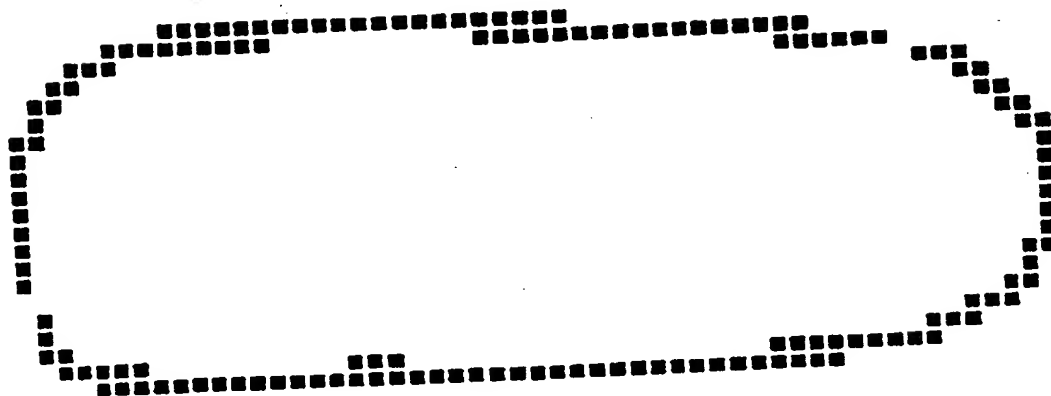


Fig. 19

Fig. 18



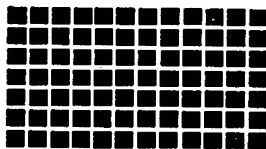
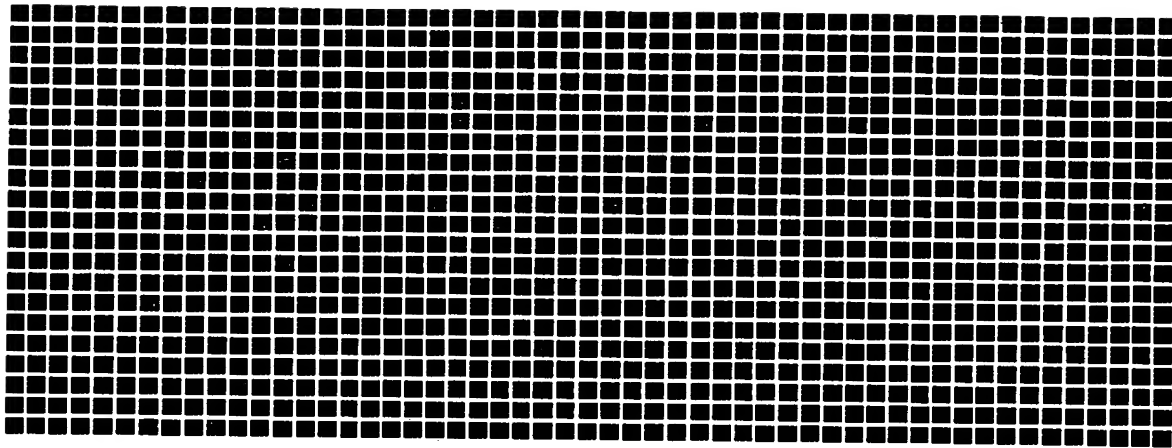


Fig. 20

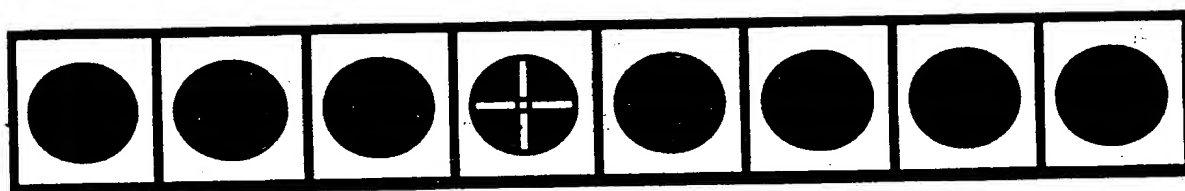


Fig. 21

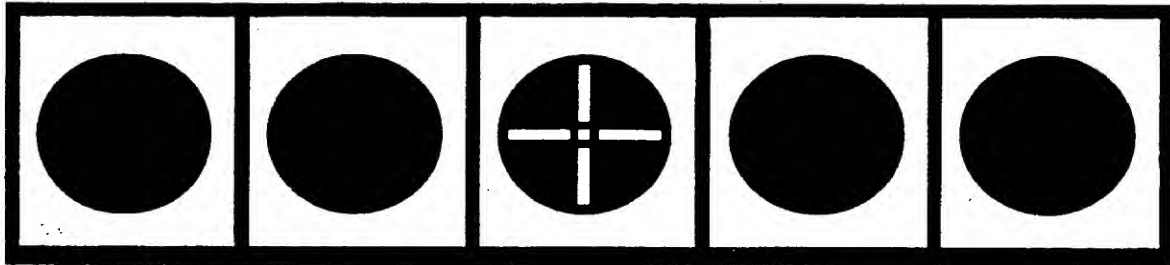


Fig. 23

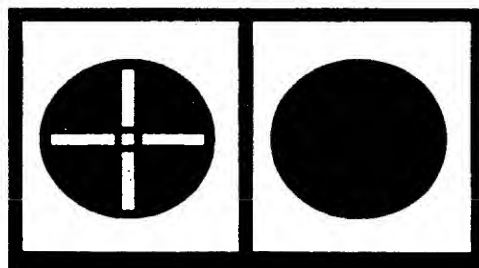


Fig. 22

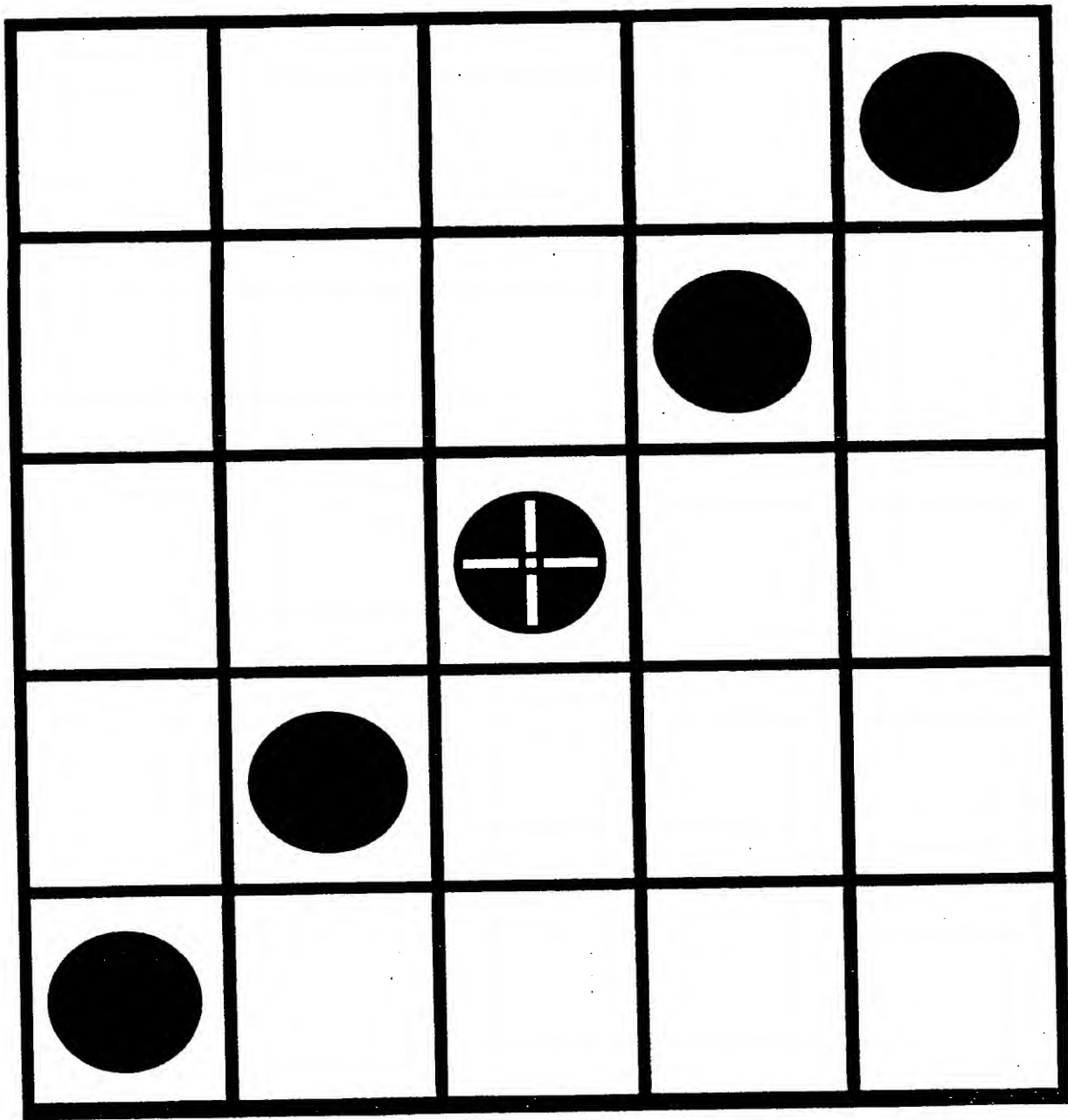


Fig. 24

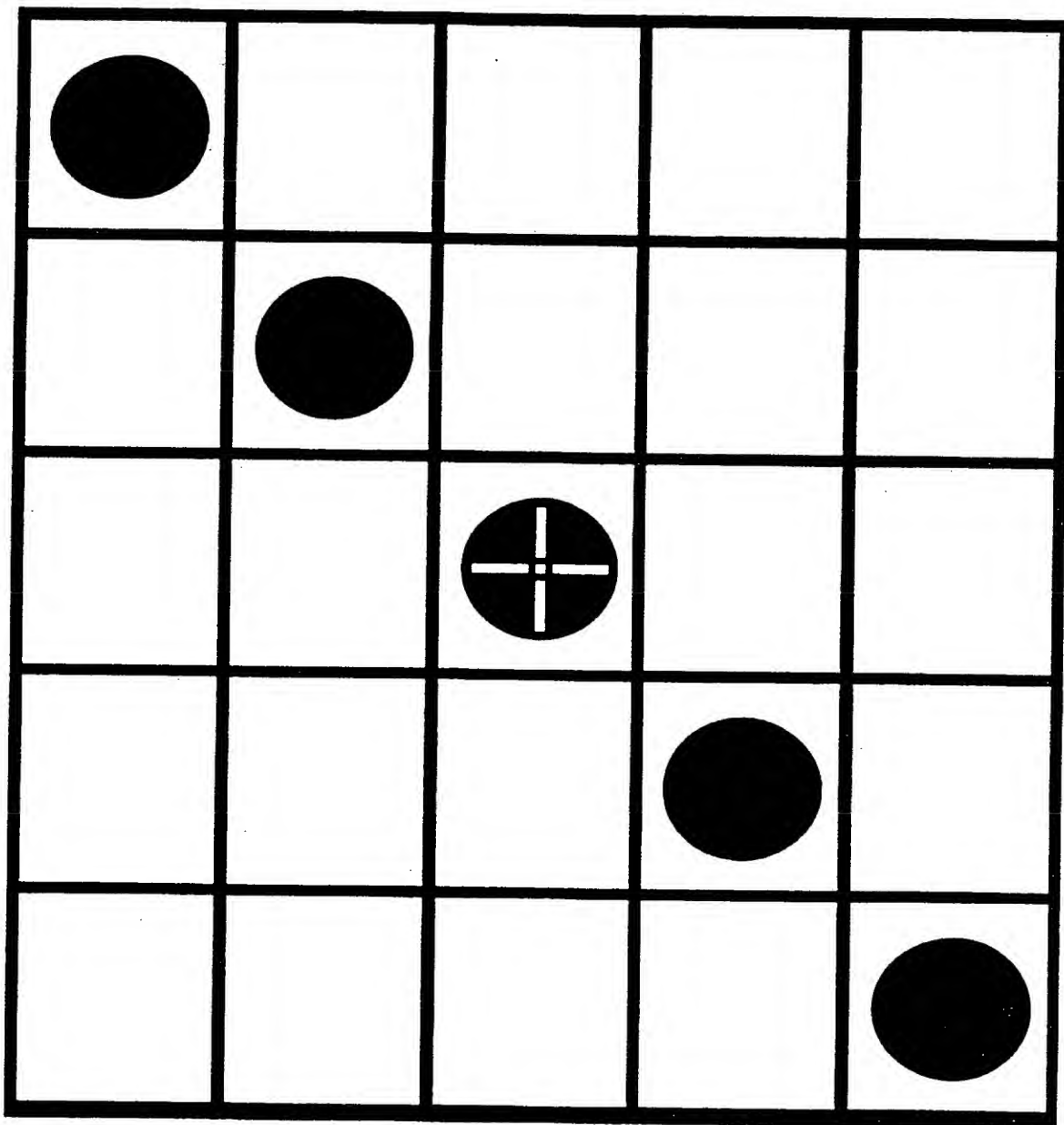


Fig. 25

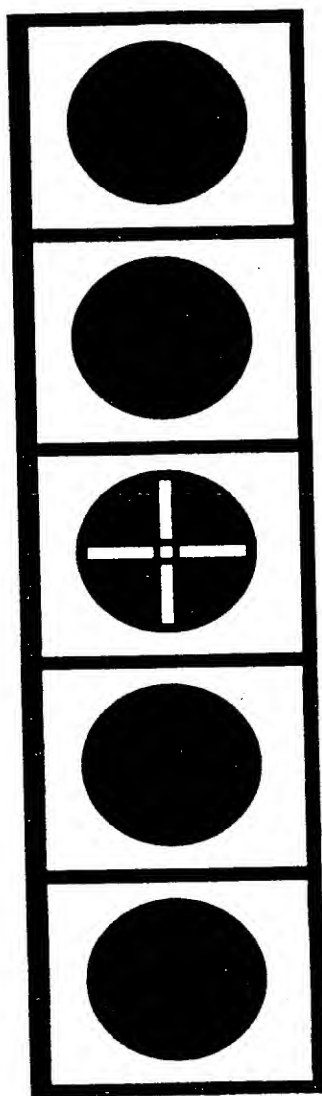
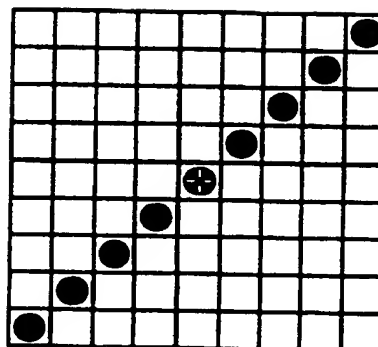


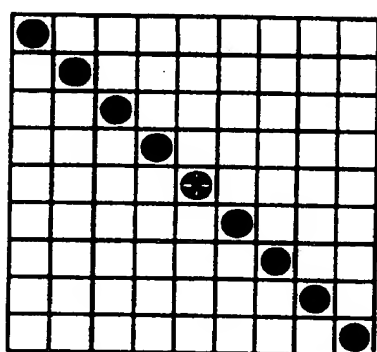
Fig. 26



A



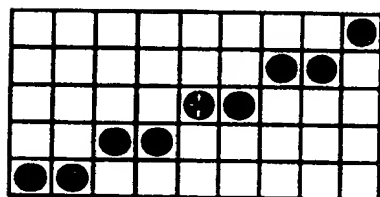
B



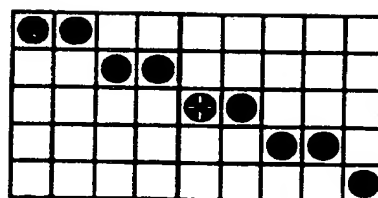
C



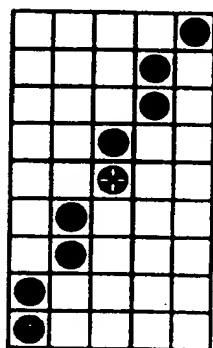
D



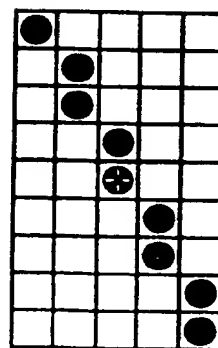
E



F



G

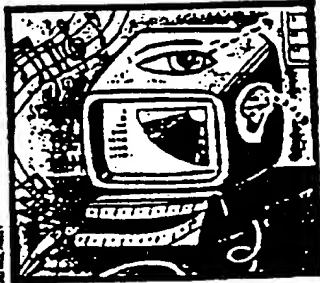


H

Fig. 27

LETTERS

Reactions and responses
from the PC World
community



Viewing Multimedia

I was impressed with Richard Landry's January column about multimedia. When I purchased my computer, the salesperson made it sound as if it would do everything. But within a year it was outdated. Landry's plain-English explanation of what's out there was a breath of fresh air. Keep the good articles coming.

Betty Bauer
Oxnard, California

Mr. Landry is correct that the PC isn't yet ready for multimedia. My Amiga 2000 came multimedia-ready for around \$2000. It captures and plays back digitized sound and attaches easily to CD ROM drives, video recorders, scanners, and musical keyboards. Don't

get me wrong: I work with DOS computers daily, and they're good machines. But give the Amiga credit where it's due. Trust me—when you see an Amiga running incredible graphics and sound with Mac and DOS applications running in background windows, you have to agree that the Amiga is to be taken seriously.

Jerry Foster
Oxford, Michigan

Tips Worth a Subscription

Over the past several years, I have noticed that computer magazines in general have started to run out of ideas for articles. Your recent issues show that you have realized the problem and done something about it. The information in January's "101 Secret Tips" was marvelous. In fact, I found the Windows 3.0 tips alone worth the price of a subscription—so I've given one to my father. I hope you continue your hands-on ideas for getting the most out of an increasingly complex array of software.

John E. Johnson, Jr.
Redwood City, California

Encouraging Piracy?

As a consultant and programmer, I was appalled to read November's *Network Q&A*. A reader requested advice on how to allow multiple users to share single-user applications on a network. He stated, "While network versions are available, upgrade charges are prohibitive." Your mild notice advising users not to violate the software license came after a description of how to violate it. This is an obvious example of a publication encouraging its readers to violate license agreements.

Michael Sullivan
Virginia Beach, Virginia

The reader's statement that he wished to recover local disk space used to store multiple copies of programs indicated that he had a copy for each user in many cases. Consumers have spent far more on multiple copies of an application than they would have on a network version. There's no reason why users should be forced to purchase network versions just to use legally registered products on their networks. We not only advised against violating license agreements, we also gave information on preventing illegal use when the number of workstations exceeds that of licenses. —Ed.

BBS File Resources

What happened to the PC Resource BBS files? Are they still available?

Theodore C. Trostle
State College, Pennsylvania

To obtain programs and utilities formerly available on the PC Resource BBS, contact Bonnie Welsh-Carroll, IDG Communications/Peterborough, 80 Elm St., Peterborough, NH 03134; 603/921-9171, 603/921-9121 (fax). Call or write Debby Bowryvull at the same address to inquire about back issues and a small inventory of special products from the *Tips, Tools & Techniques* and *Essential DOS* issues. Please be sure to include your phone number with all communications to facilitate the discussion of fees. —Ed.

Making Windows Work Better

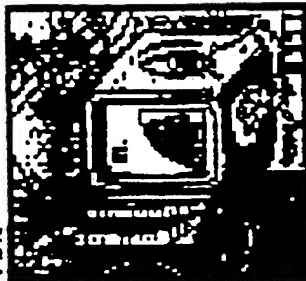
We'd like to refute the implication in November's "Thanks for the Memory" that there's no benefit in using Super PC-Kwik with Windows 3.0. Version 3.56 and later versions are all Windows (continues)

APRIL 1991 PC WORLD 35

Fig. 28

LETTERS

1. **Introduction**
 2. **Background**
 3. **Methodology**
 4. **Results**
 5. **Conclusion**
 6. **References**
 7. **Appendix**
 8. **Index**
 9. **Table of Contents**
 10. **Summary**
 11. **Abstract**
 12. **Keywords**
 13. **Subject Headings**
 14. **Notes**
 15. **References**
 16. **Appendix**
 17. **Index**
 18. **Table of Contents**
 19. **Summary**
 20. **Abstract**
 21. **Keywords**
 22. **Subject Headings**
 23. **Notes**
 24. **References**
 25. **Appendix**
 26. **Index**
 27. **Table of Contents**
 28. **Summary**
 29. **Abstract**
 30. **Keywords**
 31. **Subject Headings**
 32. **Notes**
 33. **References**
 34. **Appendix**
 35. **Index**
 36. **Table of Contents**
 37. **Summary**
 38. **Abstract**
 39. **Keywords**
 40. **Subject Headings**
 41. **Notes**
 42. **References**
 43. **Appendix**
 44. **Index**
 45. **Table of Contents**
 46. **Summary**
 47. **Abstract**
 48. **Keywords**
 49. **Subject Headings**
 50. **Notes**
 51. **References**
 52. **Appendix**
 53. **Index**
 54. **Table of Contents**
 55. **Summary**
 56. **Abstract**
 57. **Keywords**
 58. **Subject Headings**
 59. **Notes**
 60. **References**
 61. **Appendix**
 62. **Index**
 63. **Table of Contents**
 64. **Summary**
 65. **Abstract**
 66. **Keywords**
 67. **Subject Headings**
 68. **Notes**
 69. **References**
 70. **Appendix**
 71. **Index**
 72. **Table of Contents**
 73. **Summary**
 74. **Abstract**
 75. **Keywords**
 76. **Subject Headings**
 77. **Notes**
 78. **References**
 79. **Appendix**
 80. **Index**
 81. **Table of Contents**
 82. **Summary**
 83. **Abstract**
 84. **Keywords**
 85. **Subject Headings**
 86. **Notes**
 87. **References**
 88. **Appendix**
 89. **Index**
 90. **Table of Contents**
 91. **Summary**
 92. **Abstract**
 93. **Keywords**
 94. **Subject Headings**
 95. **Notes**
 96. **References**
 97. **Appendix**
 98. **Index**
 99. **Table of Contents**
 100. **Summary**
 101. **Abstract**
 102. **Keywords**
 103. **Subject Headings**
 104. **Notes**
 105. **References**
 106. **Appendix**
 107. **Index**
 108. **Table of Contents**
 109. **Summary**
 110. **Abstract**
 111. **Keywords**
 112. **Subject Headings**
 113. **Notes**
 114. **References**
 115. **Appendix**
 116. **Index**
 117. **Table of Contents**
 118. **Summary**
 119. **Abstract**
 120. **Keywords**
 121. **Subject Headings**
 122. **Notes**
 123. **References**
 124. **Appendix**
 125. **Index**
 126. **Table of Contents**
 127. **Summary**
 128. **Abstract**
 129. **Keywords**
 130. **Subject Headings**
 131. **Notes**
 132. **References**
 133. **Appendix**
 134. **Index**
 135. **Table of Contents**
 136. **Summary**
 137. **Abstract**
 138. **Keywords**
 139. **Subject Headings**
 140. **Notes**
 141. **References**
 142. **Appendix**
 143. **Index**
 144. **Table of Contents**
 145. **Summary**
 146. **Abstract**
 147. **Keywords**
 148. **Subject Headings**
 149. **Notes**
 150. **References**
 151. **Appendix**
 152. **Index**
 153. **Table of Contents**
 154. **Summary**
 155. **Abstract**
 156. **Keywords**
 157. **Subject Headings**
 158. **Notes**
 159. **References**
 160. **Appendix**
 161. **Index**
 162. **Table of Contents**
 163. **Summary**
 164. **Abstract**
 165. **Keywords**
 166. **Subject Headings**
 167. **Notes**
 168. **References**
 169. **Appendix**
 170. **Index**
 171. **Table of Contents**
 172. **Summary**
 173. **Abstract**
 174. **Keywords**
 175. **Subject Headings**
 176. **Notes**
 177. **References**
 178. **Appendix**
 179. **Index**
 180. **Table of Contents**
 181. **Summary**
 182. **Abstract**
 183. **Keywords**
 184. **Subject Headings**
 185. **Notes**
 186. **References**
 187. **Appendix**
 188. **Index**
 189. **Table of Contents**
 190. **Summary**
 191. **Abstract**
 192. **Keywords**
 193. **Subject Headings**
 194. **Notes**
 195. **References**
 196. **Appendix**
 197. **Index**
 198. **Table of Contents**
 199. **Summary**
 200. **Abstract**
 201. **Keywords**
 202. **Subject Headings**
 203. **Notes**
 204. **References**
 205. **Appendix**
 206. **Index**
 207. **Table of Contents**
 208. **Summary**
 209. **Abstract**
 210. **Keywords**
 211. **Subject Headings**
 212. **Notes**
 213. **References**
 214. **Appendix**
 215. **Index**
 216. **Table of Contents**
 217. **Summary**
 218. **Abstract**
 219. **Keywords**
 220. **Subject Headings**
 221. **Notes**
 222. **References**
 223. **Appendix**
 224. **Index**
 225. **Table of Contents**
 226. **Summary**
 227. **Abstract**
 228. **Keywords**
 229. **Subject Headings**
 230. **Notes**
 231. **References**
 232. **Appendix**
 233. **Index**
 234. **Table of Contents**
 235. **Summary**
 236. **Abstract**
 237. **Keywords**
 238. **Subject Headings**
 239. **Notes**
 240. **References**
 241. **Appendix**
 242. **Index**
 243. **Table of Contents**
 244. **Summary**
 245. **Abstract**
 246. **Keywords**
 247. **Subject Headings**
 248. **Notes**
 249. **References**
 250. **Appendix**
 251. **Index**
 252. **Table of Contents**
 253. **Summary**
 254. **Abstract**

[illegible]

ANALYST

Consent, Dated: _____

The following information is being
 furnished to you for your information only.
 This is not a recommendation or
 an offer to buy or sell any security or
 to participate in any transaction, and
 it does not constitute an offer to
 sell or a solicitation to buy any security.
 The information is being furnished to you
 for your information only.

[illegible]

Donny Osmond
Dancing the Night Away

[illegible]

உதவி செய்வதற்கு
அருள் செய்து கொடுக்க

1488 2017 10 10 10 10 10

[illegible]

၂၀၁၁ ခုနှစ်
 ဇွန်လ ၁ ရက်

[illegible]

WALL IS BEING USED TO THE FC SYSTEM
THE FIRST AND ONLY ONE OF THE
TWO IN THE TRAIL
JULY 1968 PHOTOGRAPH

[illegible]

1. The first step is to identify the problem. In this case, the problem is that the system is not working properly.

41 45 48 52 55

Fig. 29

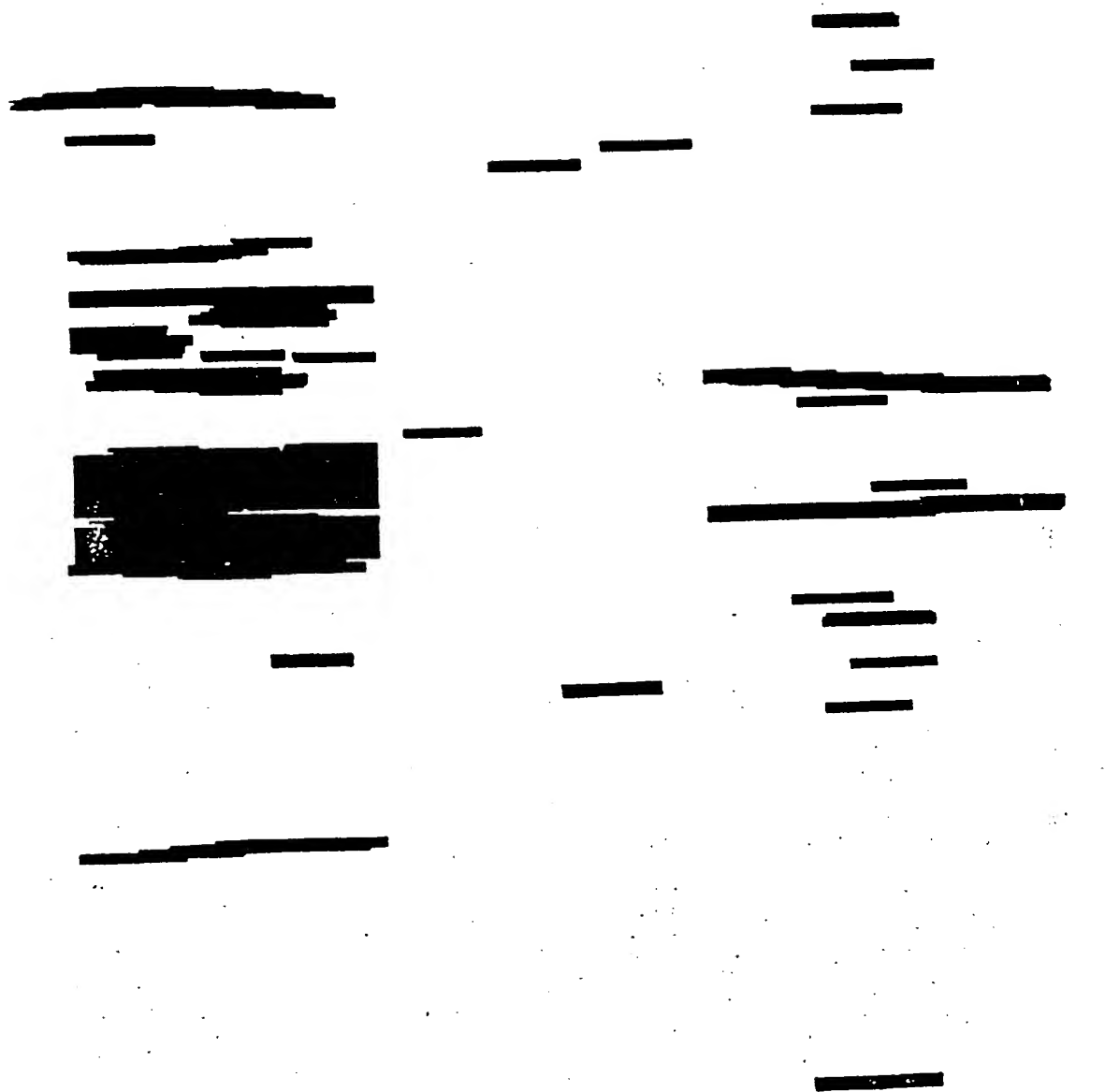


Fig. 30

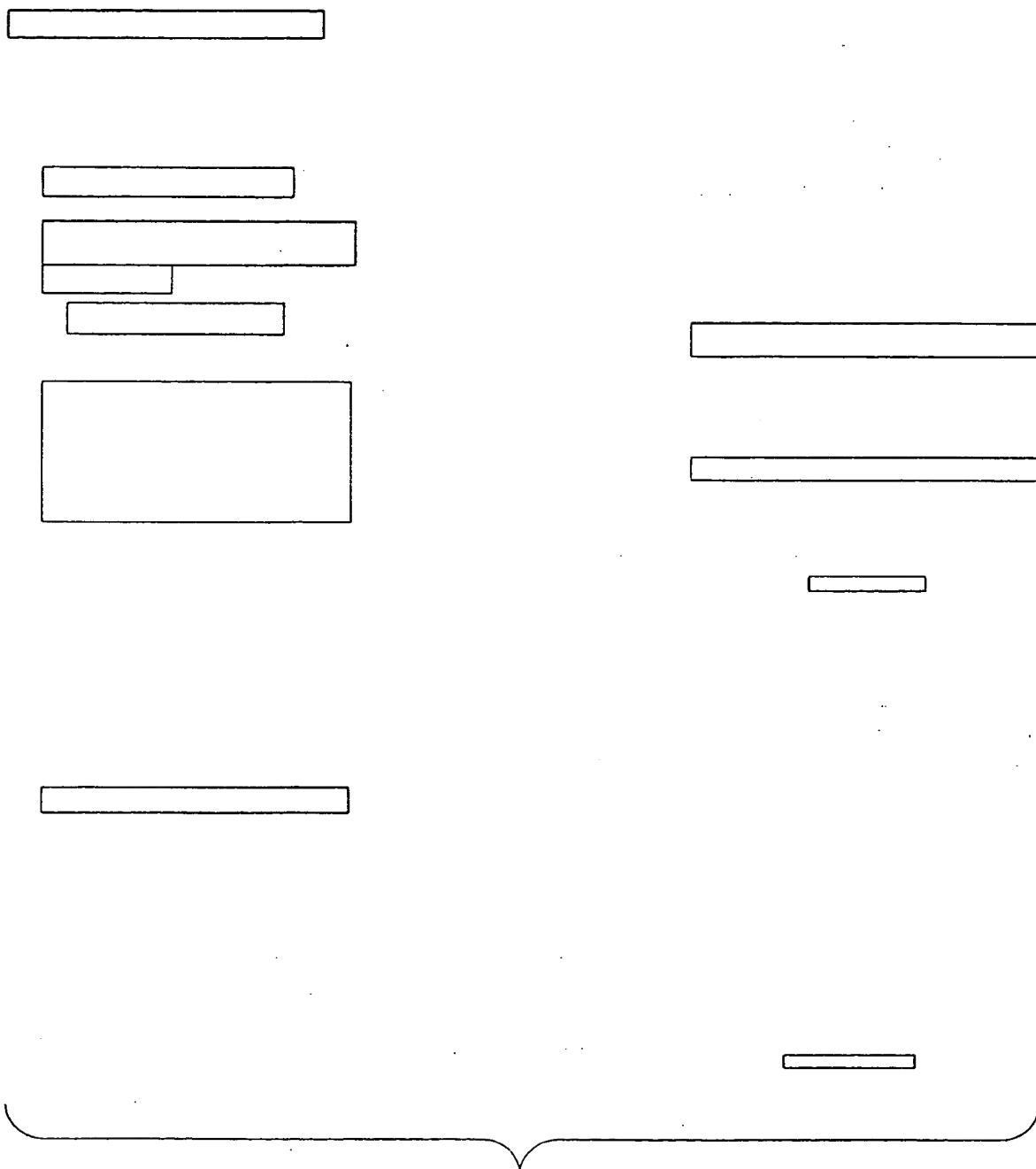


Fig. 31

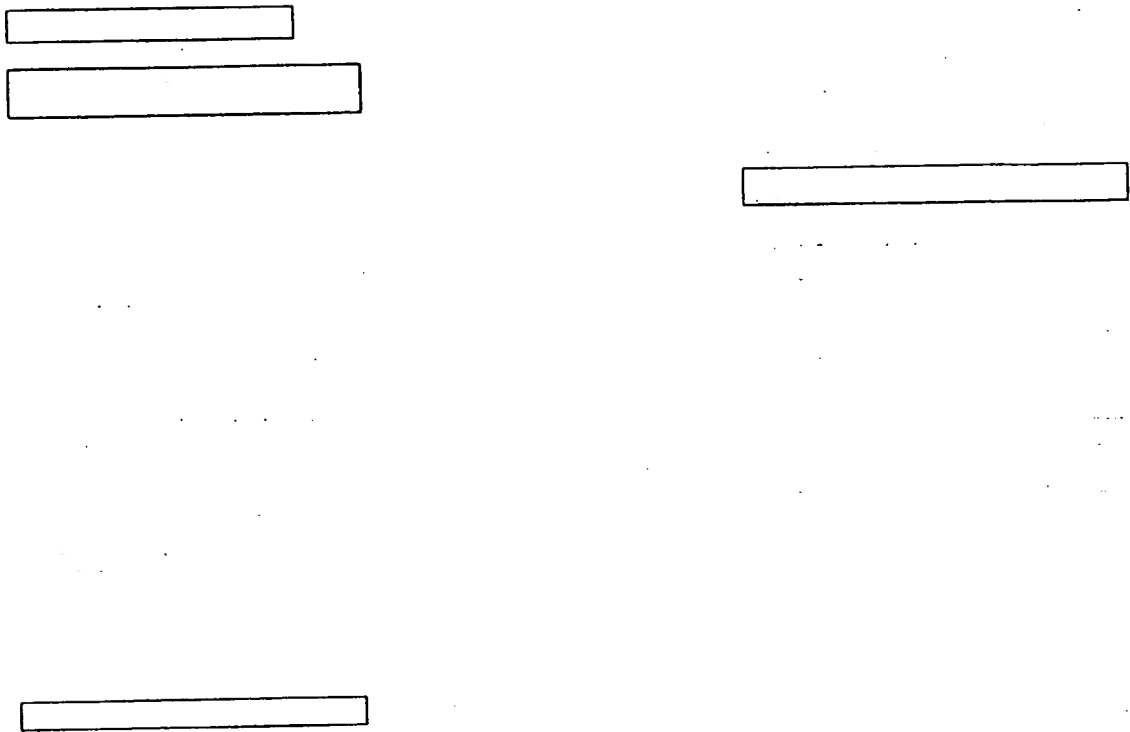


Fig. 32

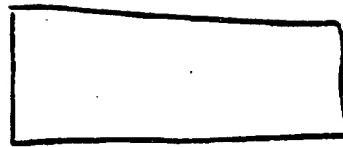
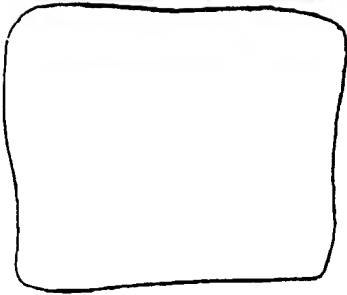
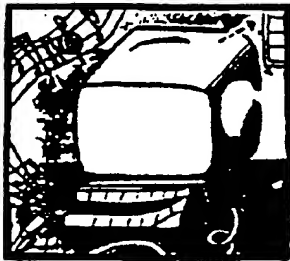
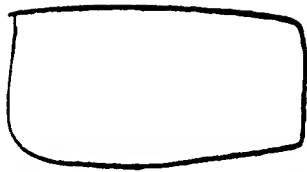
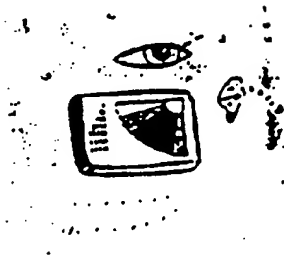


Fig. 33

actions and responses
from the PC World
community



Viewing Multimedia
I was impressed with Richard Landry's
January column about multimedia.
When I purchased my computer, the
salesperson made it sound as if it would
do everything. But within a year it was
outdated. Landry's plain-English ex-
planation of what's out there was a
breath of fresh air. Keep the good arti-
cles coming.

Betty Bauer
Oakland, California

BBS File Resources
What happened to the PC Resource
BBS files? Are they still available?
Theodore C. Tronis
State College, Pennsylvania

Fig. 34

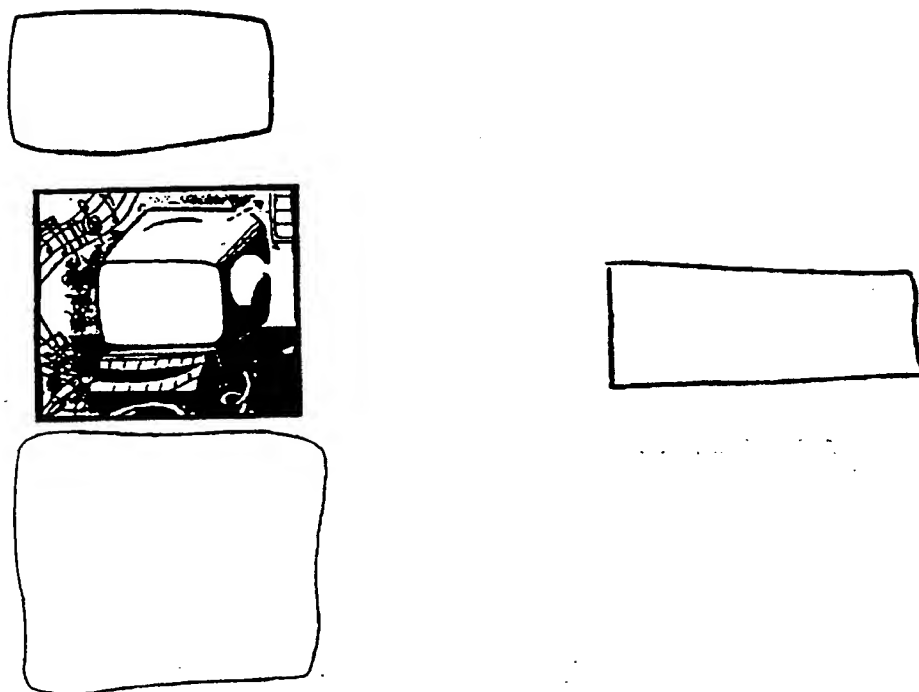


Fig. 35

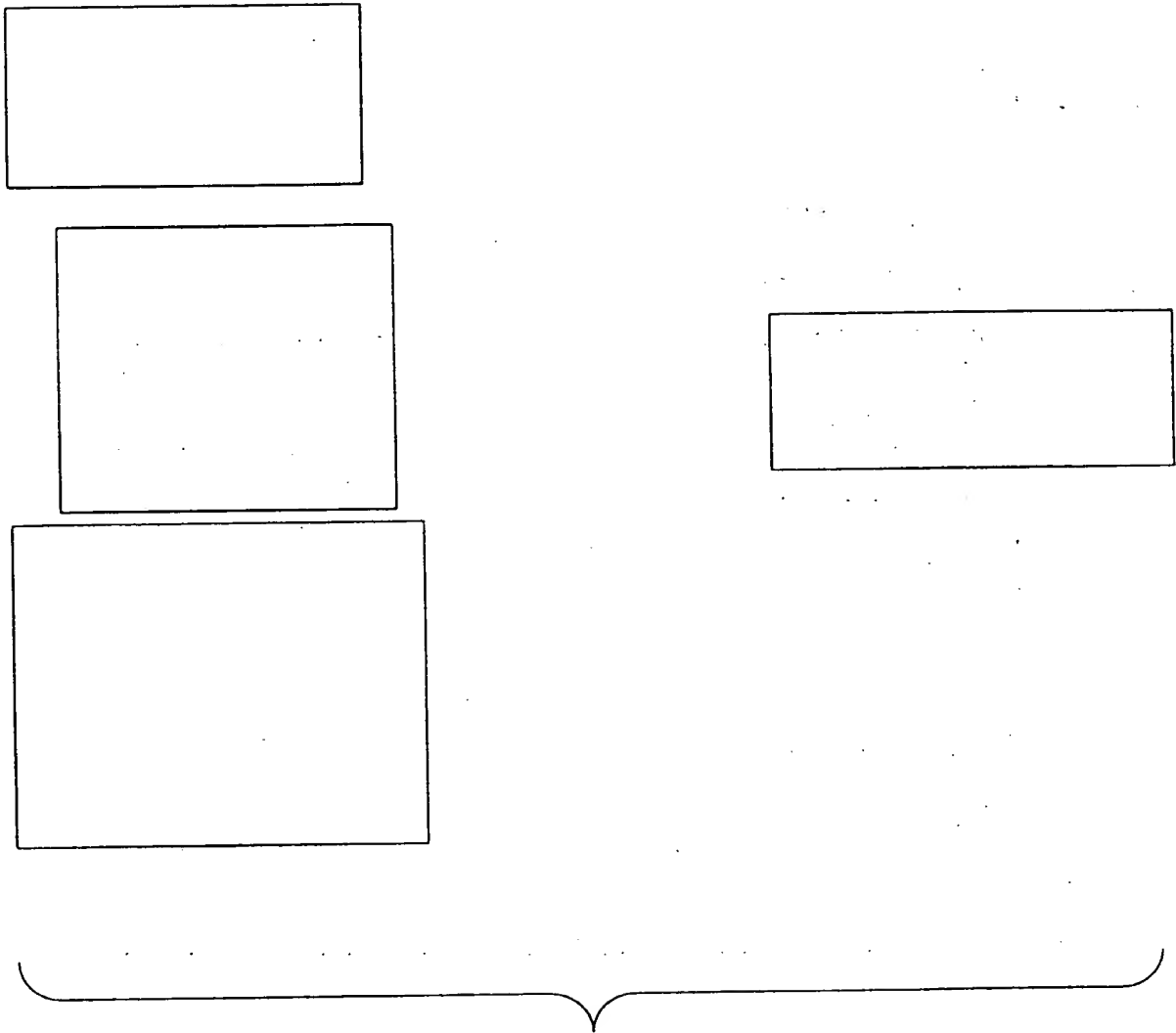
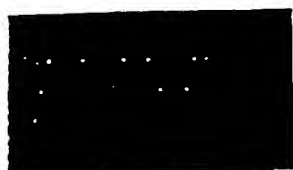
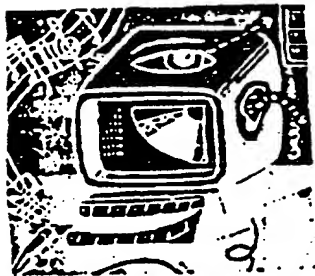


Fig. 36



Viewing Multimedia
 I was impressed with Richard Landry's January column about multimedia. When I purchased my computer, the salesperson made it sound as if it would do everything. But within a year it was outdated. Landry's plain-English explanation of what's out there was a breath of fresh air. Keep the good articles coming.
Betty Bauer
 Oxnard, California

Fig. 37

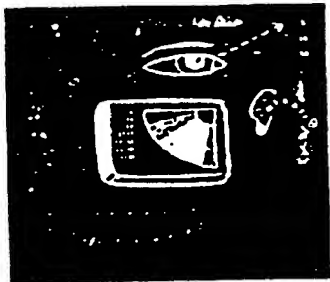


Viewing Multimedia

I was impressed with Richard Landry's January column about multimedia. When I purchased my computer, the salesperson made it sound as if it would do everything. But within a year it was outdated. Landry's plain-English explanation of what's out there was a breath of fresh air. Keep the good articles coming.

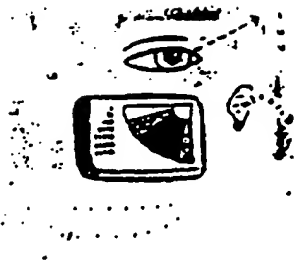
*Betty Bauer
Oxnard, California*

Fig. 38



Viewing Multimedia
 I was impressed with Richard Landry's January column about multimedia. When I purchased my computer, the salesperson made it sound as if it would do everything. But within a year it was outdated. Landry's plain-English explanation of what's out there was a breath of fresh air. Keep the good articles coming.
Betty Bauer
Oxnard, California.

Fig. 39



Viewing Multimedia

I was impressed with Richard Landry's January column about multimedia. When I purchased my computer, the salesperson made it sound as if it would do everything. But within a year it was outdated. Landry's plain-English explanation of what's out there was a breath of fresh air. Keep the good articles coming.

Betty Bauer
 Ormond, California

Fig. 40

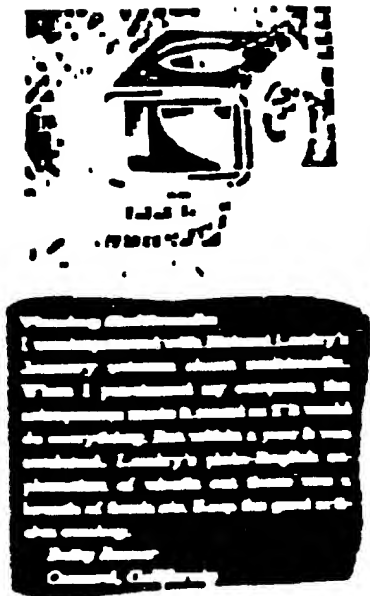


Fig. 41

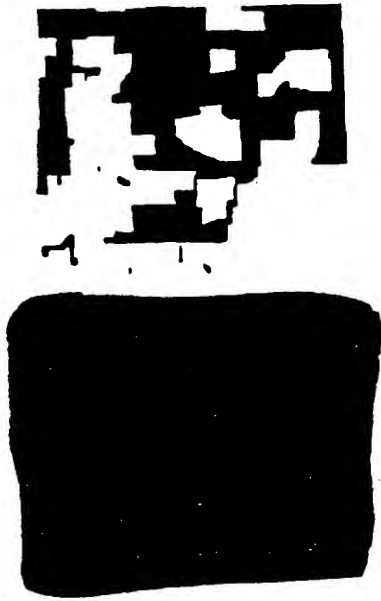


Fig. 42



Viewing Multimedia

I was impressed with Richard Landry's January column about multimedia. When I purchased my computer, the salesperson made it sound as if it would do everything. But within a year it was outdated. Landry's plain-English explanation of what's out there was a breath of fresh air. Keep the good articles coming.

*Betty Bauer
Ormond, California*

Fig. 43

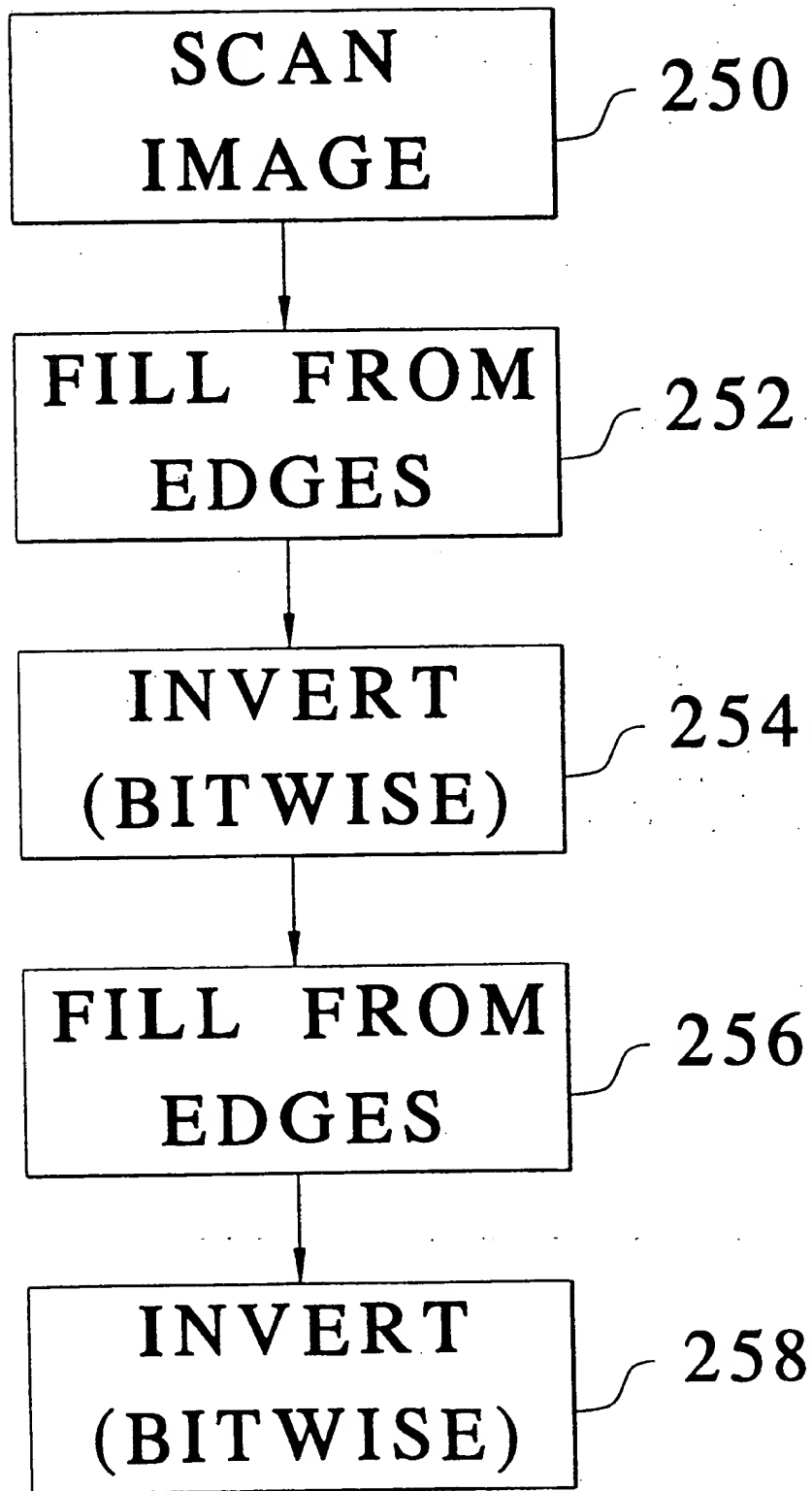


FIG. 44

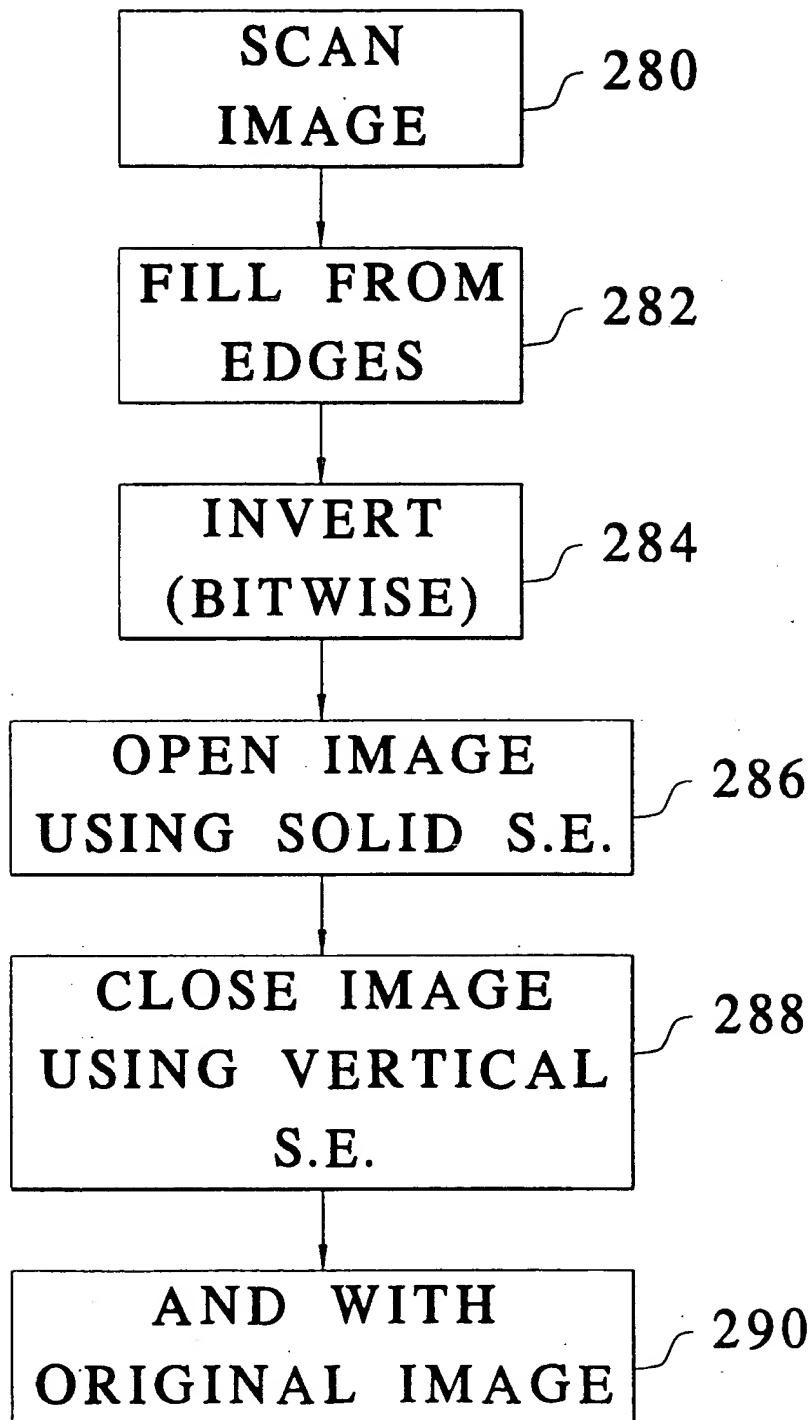


FIG. 45

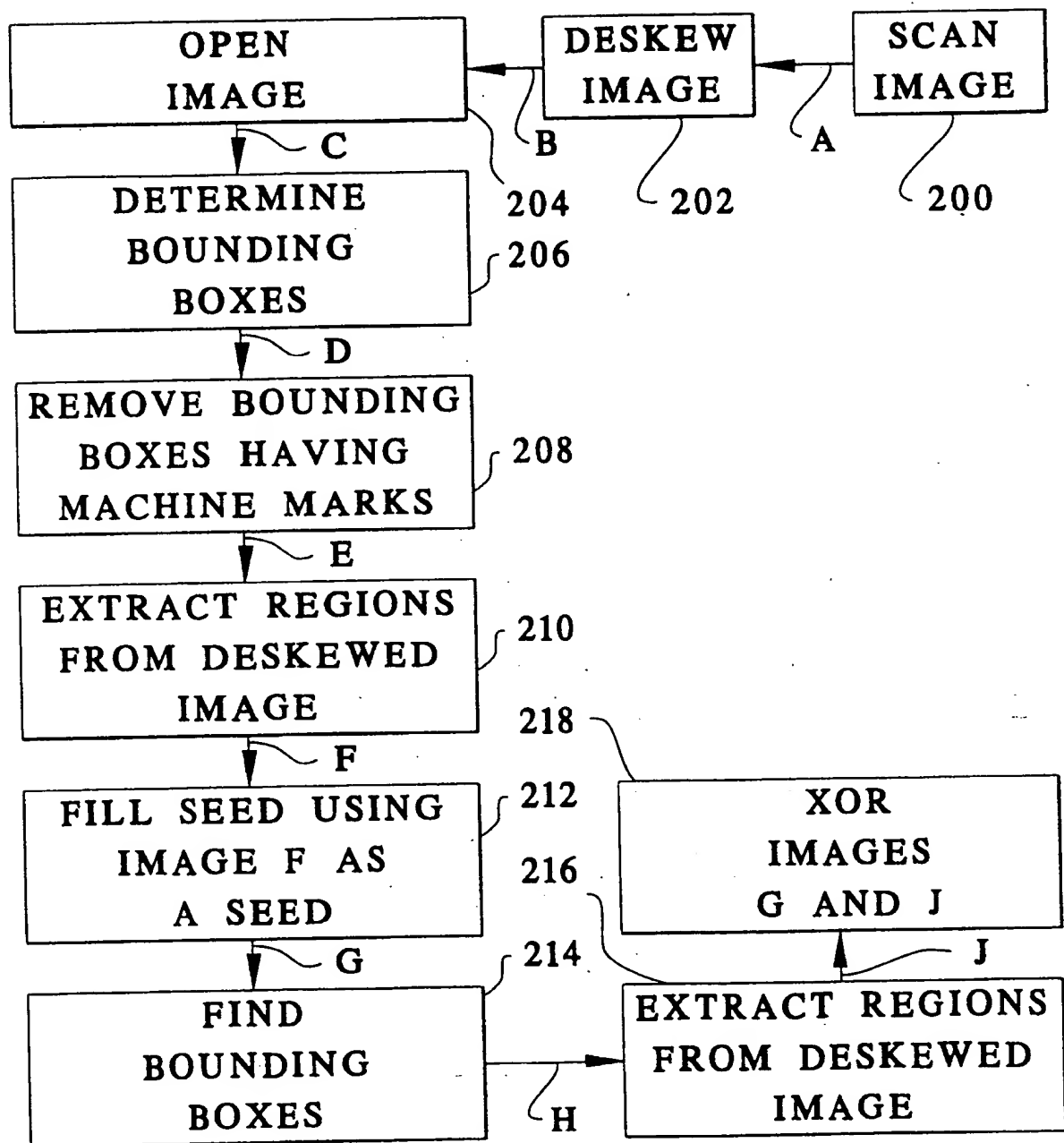
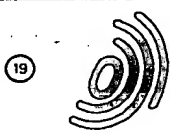


FIG. 46

THIS PAGE BLANK (USPTO)



Europäisches Patentamt
European Patent Office
Office européen des brevets



Publication number: 0 543 599 A3

(12)

EUROPEAN PATENT APPLICATION

(21) Application number: 92310438.4

(51) Int. Cl.⁵: G06K 9/20

(22) Date of filing: 16.11.92

(30) Priority: 19.11.91 US 794275

(43) Date of publication of application:
26.05.93 Bulletin 93/21

(84) Designated Contracting States:
DE FR GB

(88) Date of deferred publication of the search report:
22.12.93 Bulletin 93/51

(71) Applicant: XEROX CORPORATION
Xerox Square
Rochester New York 14644(US)

(72) Inventor: Bloomberg, Dan S.
1013 Paradise Lane
Palo Alto, California 94306(US)
Inventor: Lakshmi, Dasari
3375 Alma Street No. 383
Palo Alto, California 94304(US)

(74) Representative: Johnson, Reginald George et al
Rank Xerox Patent Department,
Albion House,
55-59 New Oxford Street
London WC1A 1BS (GB)

(54) Method and apparatus for image hand markup detection.

(57) An image markup detection device and method identifies and extracts markup lines and regions marked automatically or interactively by a user with an ordinary pen or pencil. Only morphological image processing operations on a scanned source image are used, resulting in the extrapolation of markup lines and marked regions. The markup lines are either extracted from the image, or the background information of the image (e.g., text) is removed, leaving only the markup lines. The marked region can then be printed, transferred or otherwise processed.

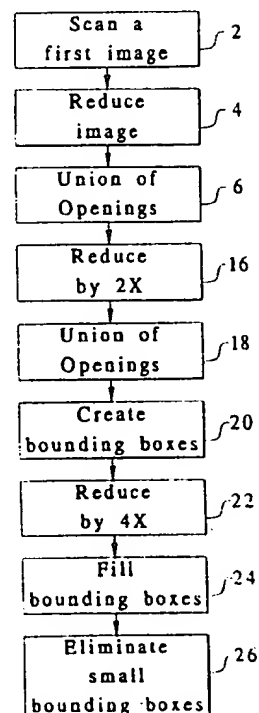


Fig. 2

EP 0 543 599 A3



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number

EP 92 31 0438

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl. 5)
X	US-A-5 048 099 (LEE YONGCHUN) 10 September 1991	1,10	G 06 K 9/20
Y	Idem * column 2, line 39 - line 64; figures 1,5,10; column 5, line 39 - line 68 *	3,2	
Y	US-A-5 048 109 (BLOOMBERG DAN S ET AL) 10 September 1991	2,3	
A	C.R. GIARDINA & E.R. DOUGHERTY 'Morphological methods in image and signal processing' 1988 , PRENTICE HALL, INC. , ENGLEWOOD CLIFFS, US * page 111 - page 125; Ch. 4 Topological processing *	5-9	
A	OPTICAL ENGINEERING vol. 26, no. 7 , July 1987 , BELLINGHAM US pages 623 - 632 P. MARAGOS 'Tutorial on advances in morphological image processing and analysis' * the whole document *	1-10	TECHNICAL FIELDS SEARCHED (Int. Cl.5)
A	SYSTEMS & COMPUTERS IN JAPAN vol. 17, no. 3 , 1986 , NEW YORK US pages 35 - 46 Y. SUENAGA 'Some Techniques for Document and Image Preparation'		G 06 K
The present search report has been drawn up for all claims			
Place of search THE HAGUE		Date of completion of the search 22-10-1993	Examiner VAN GINKEL K E A
CATEGORY OF CITED DOCUMENTS			
X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document		I : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application F : document cited for other reasons A : member of the same patent family, corresponding document	